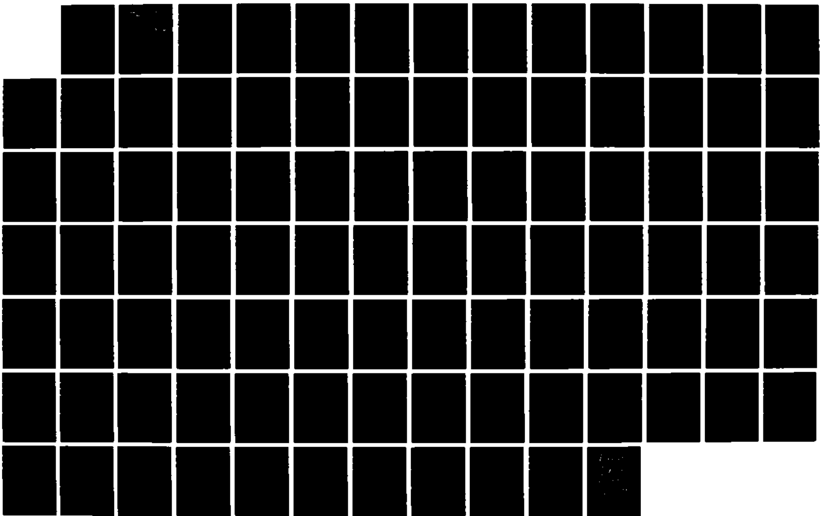
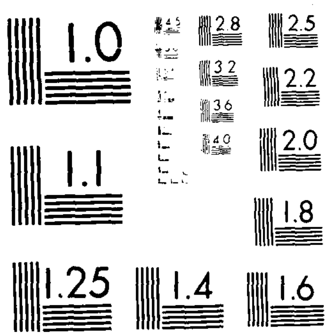
AD-A192 069 AN INTERACTIVE COMPUTER CODE FOR PRELIMINARY DESIGN OF 1/1
SOLID PROPELLANT ROCKET MOTORS(U) NAVAL POSTGRADUATE
SCHOOL MONTEREY CA C YUAN DEC 87

UNCLASSIFIED F/G 21/8.2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

2

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

AN INTERACTIVE COMPUTER CODE FOR
PRELIMINARY DESIGN
OF SOLID PROPELLANT ROCKET MOTORS

by

Chung-I. YUAN

December 1987

Thesis Advisor: D. W. NETZER

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL (If applicable) 67	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification) UNCLASSIFIED
AN INTERACTIVE COMPUTER CODE FOR PRELIMINARY DESIGN OF SOLID PROPELLANT ROCKET MOTORS. UNCLASSIFIED

12. PERSONAL AUTHOR(S)
YUAN, CHUNG-I.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
MASTER'S THESIS	FROM ______ TO ______	1987 DECEMBER	92

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	solid propellant rocket motor design, rocket motor

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

An interactive computer code for the preliminary design of solid propellant rocket motors("SPRMD") was successfully developed and its use was demonstrated through a design example. "SPRMD" was written in FORTRAN for use on an IBM PC/AT. It combined several existing codes ("MICROPEP", "GRAINS", "ROCKET", etc.) and used the performance loss estimation methods suggested by the AGARD Proppulsion and Energetics Panel for aluminized propellants.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL D.W. NETZER	22b. TELEPHONE (Include Area Code) 408/646-2980 22c. OFFICE SYMBOL 67Nt

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

☆ U.S. Government Printing Office: 1986—606-24.

UNCLASSIFIED

An Interactive Computer Code for Preliminary Design of Solid Propellant Rocket Motors

by

Chung-I. Yuan
Lieutenant Commander, Republic of China Navy
B.S., Chinese Naval Academy, Republic of China, 1976

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1987

Author: Chung-I. Yuan
Chung-I. Yuan

Approved by: David W. Netzer
D. W. NETZER, Thesis Advisor

Max F. Platzer
M. F. PLATZER, Chairman
Department of Aeronautics

G. E. Schacher
G. E. SCHACHER, Dean
Science and Engineering

ABSTRACT

An interactive computer code for the preliminary design of solid propellant rocket motors ("SPRMD") was successfully developed and its use was demonstrated through a design example. "SPRMD" was written in FORTRAN for use on an IBM PC/AT. It combined several existing codes ("MICROPEP", "GRAINS", "ROCKET", etc.) and used the performance loss estimation methods suggested by the AGARD Propulsion and Energetics Panel for aluminized propellants.

ACKNOWLEDGMENTS

I would like to express my sincere appreciation to Professor D. W. Netzer for his professional guidance as well as patience in support of this thesis research on solid rocket motor design. The thesis provided me the opportunity to work on the topic which interested me most in all of my courses of MSAE study at the Naval Postgraduate School.

My deep gratitude also goes to my wife, Tang-Ming, for her encouragement, support, and forgiving of my ignorance of many things in daily life, such as our anniversary and her birthday.

TABLE OF CONTENTS

TABLES

FIGURES

I. INTRODUCTION

A. BACKGROUND

Solid propellant rocket motor design is a very complex process which often requires many iterations in order to accomplish a successful design. Usually, the first step in the design process it to obtain accurate input data on missile system restraints. Examples of these restraints are ballistic requirements (which can include thrust, burning time, total impulse) and mission/vehicle requirements (which can include motor length, diameter, allowed hazard and plume properties, altitude, etc.). These parameters are often referred to as independent parameters. Through the use of these independent parameters, the required propellant burn time and properties can be roughly estimated. Then, theoretical motor performance and losses can be estimated. The thrust-time requirements together with the expected structural loads permit initial selection of a grain design.

The next step of solid propellant rocket motor design is to use the propellant properties and grain design to predict the pressure- and thrust-time profiles and the total impulse of the motor. Propellant properties and grain design are then iterated until these independent variable requirements are met, together with acceptable propellant stress levels. A final step in the missile design process is to use the

thrust-time profile in an external aerodynamics code to ensure that range and terminal velocity requirements are met.

During the past decade, there has been significant progress in the automation of solid rocket motor design. It is mainly because of the extensive use of digital computers, which greatly facilitate the iterative type computations mentioned above (Ref. 1). In recent years, many of the design processes have been accomplished using separate computer codes, i.e., adiabatic equilibrium combustion codes, codes which estimate performance losses, grain geometry optimization codes, etc. These codes have, in some cases, been combined into much larger design codes, and they are often difficult and expensive to run. Recently, PC versions of some rocket motor design codes have become available. However, they must still be used individually, and were not all available for use on the same computer. The use of these codes was also very inconvenient in that the same parameters often had different variable names in the various codes. The user had to transfer or convert the data from one code to the next.

In addition, a recent AGARD publication has provided recommended emirical equations which can be used for rapid and accurate estimation of performance losses for aluminized propellants.

B. OBJECTIVES

The first objective was to outline a suitable "Solid Propellant Rocket Motor Preliminary Design Procedure" which, of course, is not a unique procedure. The second objective was to create, modify, and/or combine the PC versions of the necessary design codes into one interactive code which could be conveniently used for preliminary design on an IBM PC/AT microcomputer. The final objective was to demonstrate the program by designing a tactical solid propellant rocket motor combustor.

C. APPROACH

The approach to achieve the objectives was first to detail an acceptable preliminary design process. The algorithm of the design process was then followed, utilizing several existing codes and loss estimation methods as appropriate, to obtain an integrated design code for use on the IBM PC/AT. The programs to be used were (1) "MICROPEP" (a microcomputer version of a propellant evaluation program from China Lake, Naval Weapons Center), which calculates the adiabatic, equilibrium combustion process, and theoretical motor performance; (2) "GRAINS" (a radial spoked star grain design program from CSD); (3) "ROCKET (a Lockheed internal ballistics program), which is a one-dimensional flow code which predicts pressure and thrust versus time; and (4) "FLYIT" (a flight simulation program from China Lake, Naval

Weapons Center), which performs simplified trajectory calculations. In addition, a code based upon AGARD-AR-230 was written for estimation of performance losses for aluminized propellants. Capabilities of the final program were then to be demonstrated by considering the design of a tactical motor.

II. SOLID PROPELLANT ROCKET MOTOR DESIGN PROCESS

A. GENERAL

The method of solid rocket motor design adopted in this work is shown in Figure 1. It was assumed that the required information on fundamental propellant properties (burning rates, temperature sensitivity, etc.) was known. In addition, the design process discussed herein did not include any stress analysis for the propellant grain or case.

B. INPUTS

The inputs that are required to satisfy mission objectives (the independent parameters) were as follows:

1. Ballistics Performance Requirements

   a. Average thrust ($\overline{F}$) or thrust-time profiles,

   b. Burn time (t_b), or

   c. Total impulse (I_t).

   These parameters are interrelated as shown in Figure 2. Thus

$$I_t = \int_0^{t_b} F\,dt \tag{2.1}$$

$$\overline{F} = I_t / t_b \tag{2.2}$$

2. Mission/Vehicle Constraints

   These requirements are directly related to the motor design and include:

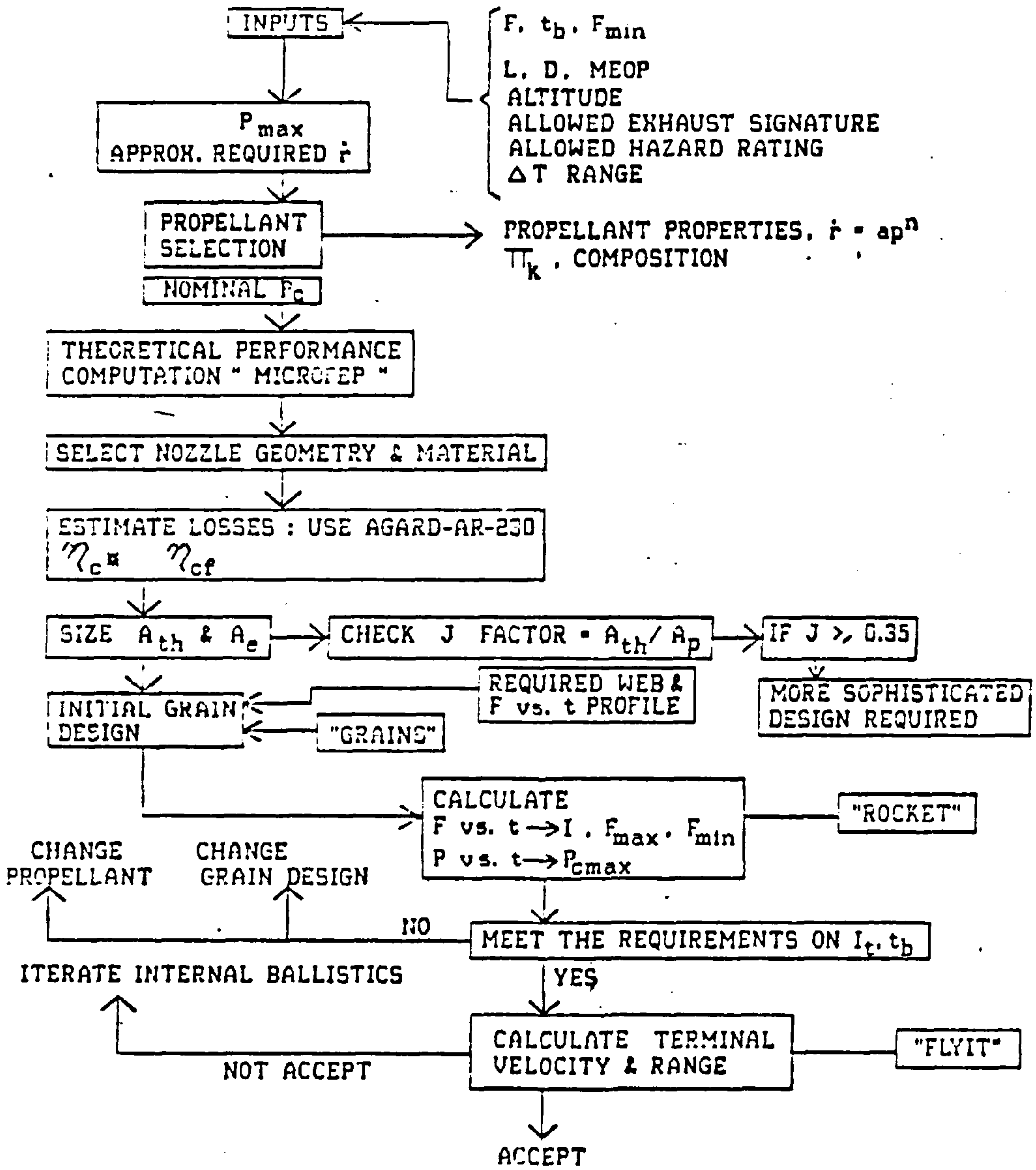


FIGURE 1.

DESIGN PROCESS

a. Envelope

The allowable envelope (motor length L, motor inside diameter D) is the fundamental constraint on the motor geometry, which can be used to estimate the required burning rate ($\dot{r}$) and total volume available (V_a) for the propellant by following relations:

$$\dot{r} \cong (D/4)/t_b \qquad (2.3)$$

(i.e., a web fraction of approximately .5)

$$Va = (\pi D^2/4)\ L \qquad (2.4)$$

b. Maximum Expected Operating Pressure (MEOP)

The MEOP is set by the structural limit of the motor case. The maximum operating pressure at maximum propellant temperature (P_{maxtp}) is less than the MEOP by the factor of reproducibility tolerance (see Figure 2). P_{maxtp} can be initially estimated from the following equation

$$P_{maxtp} = \text{MEOP (1-reproducibility tolerance)} \qquad (2.5)$$

c. Temperature Range and Altitude

The operating propellant temperature range effects the selection of the propellant through the required burning rate temperature sensitivity. The design altitude, together with the nominal chamber pressure determine the exhaust nozzle area ratio.

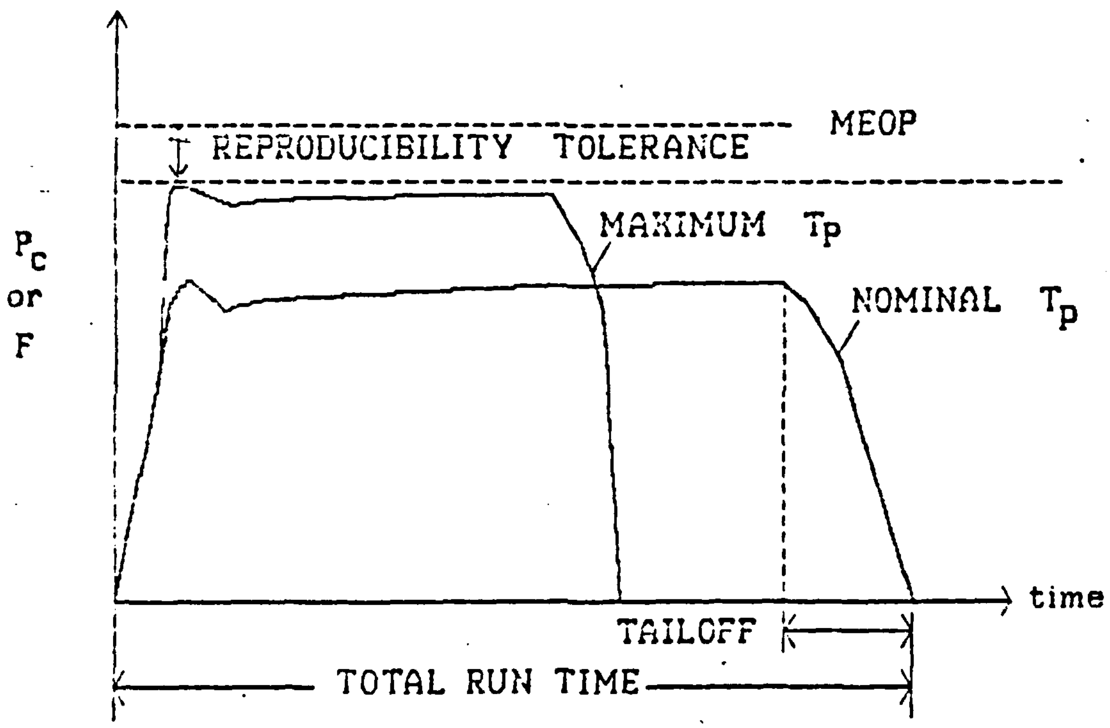


FIGURE 2.

TYPICAL PRESSURE OR THRUST VERSUS TIME PROFILES

d. Allowed Hazard Rating and Allowed Exhaust Signature

The hazard rating requirements have to do with whether the propellant is detonable or non-detonable. Modern high energy propellants are often detonable. The allowed exhaust signature affects the propellant selection through requirements on exhaust smoke and radiation.

C. PROPELLANT SELECTION

From the ballistics performance and mission/vehicle constraints, it is possible to make a preliminary selection of the propellant type that is most likely to give the required performance, internal ballistics, flame temperature and mechanical properties, as well as the necessary storage stability, the best hazard properties and exhaust signature (Ref. 2). A particular propellant will have known (or estimated) burning rate ($\dot{r}$), temperature sensitivity (π_K), density (ρ_p), and composition. Then the nominal chamber pressure ($P_{c_{nom}}$) can be calculated by using the equation (Ref. 3):

$$P_{c_{nom}} = P_{maxtp} \exp (\pi_K (T_{nom} - T_{max})) \tag{2.6}$$

D. THEORETICAL PERFORMANCE COMPUTATION

Following the propellant selection, theoretical performance computations can be made for the specified chamber and ambient pressures. A program such as PEPCODE (Ref. 4),

or the PC version "MICROPEP", will provide data such as thrust coefficient, characteristic exhaust velocity, shifting and frozen equilibrium performance results, optimum nozzle expansion ratio, mole fraction of condensed species in the exhaust, etc.

E. NOZZLE GEOMETRY AND MATERIAL SELECTION

The information on theoretical combustion temperature, pressure, gas composition, and burn time, together with geometry constraints, permits the nozzle configuration and material to be selected.

F. PERFORMANCE LOSS ESTIMATION

It was assumed that the delivered specific impulse could be written:

$$I_{sp} = I_{sp_{th}} \, \eta_{C_f} \, \eta_{C^*} \tag{2.7}$$

where $I_{sp_{th}}$ is the theoretical specific impulse calculated by "MICROPEP"; and η_{C_f} and η_{C^*} are the thrust coefficient efficiency and characteristic velocity efficiency (or combustion efficiency), respectively.

η_{C_f} was computed by summing the effects from the following losses

1. Divergence losses (ε_{DIV}),
2. Two-phase flow losses (ε_{TP}),
3. Boundary layer loss (ε_{BL}),

4. Kinetics loss (ε_{KIN}),

5. Submergence loss (ε_{SUB}),

6. Erosion loss (ε_{EROS}) .

Thus, $\eta_{C_f} = 1-(\varepsilon_{DIV}+\varepsilon_{TP}+\varepsilon_{BL}+\varepsilon_{KIN}+\varepsilon_{SUB}+\varepsilon_{EROS})/100$ (2.8)

The combustion efficiency depends primarily on the residence time (t_r), and the latter can be expressed as

$$t_r = (Vol/\dot{m})\ (P_c\ \overline{M}/12\ \overline{R}\ T_c) \qquad (2.9)$$

where

- Vol is the instantaneous volume of the combusor (in^3),
- P_c is the chamber pressure (lbf/in^2),
- T_c is the chamber temperature (^0R),
- $\overline{M}$ is the molecular weight,
- $\overline{R}$ is the universal gas constant ($ft\ lbf/lb_{mole}\,^0R$),and
- $\dot{m}$ is the mass discharge rate (lbm/sec).

For example, Figure 3 (Ref. 2) is a plot of specific impulse versus average residence time for aluminized propellants. It shows that the residence time must be somewhere between 10 to 15 msec to obtain a combustion efficiency between 90% and 95%. Increasing the free volume of the combustion chamber will improve the residence time. In this work the initial volume was used in the residence time calculation, i.e., the worst case.

G. THROAT AREA AND EXIT AREA

Accurately sizing the throat and exit areas is one of the major goals of the solid propellant rocket motor design.

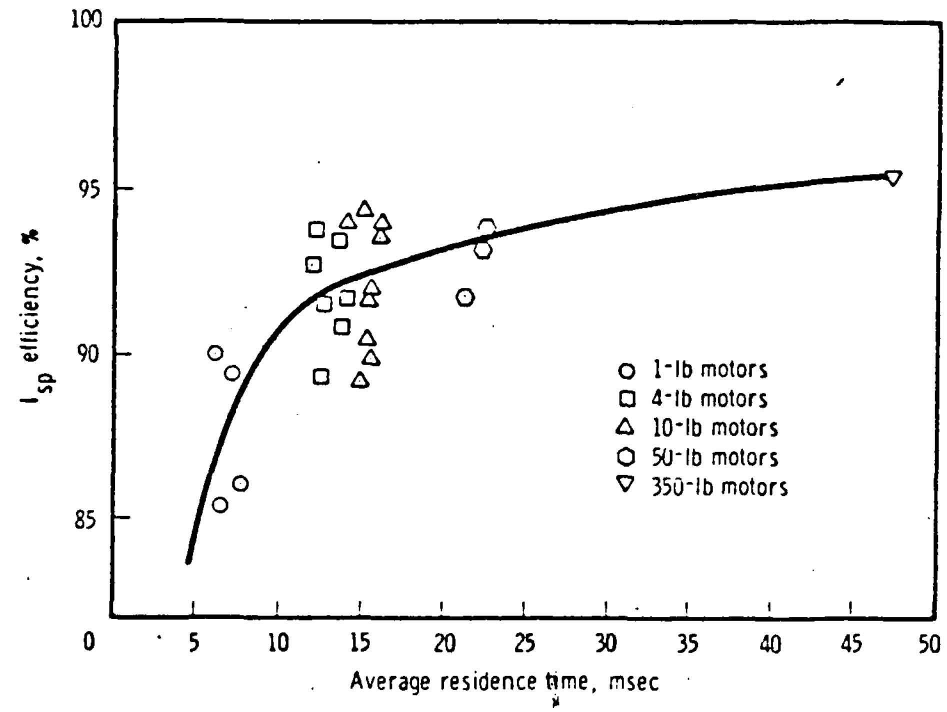


FIGURE 3.

EFFECT OF RESIDENCE TIME ON I_{sp} EFFICIENCY (Ref 2)

These areas are dependent on several other variables, such as thrust coefficient efficiency, combustion efficiency, specific impulse, and etc.. In the sizing process, it is convenient to start by using the theoretical values. The effects of losses can then be handled iteratively until the solution converges to the final areas. The rest of this section will present the basic algorithm for the throat and exit area sizing process.

The basic definition of specific impulse can be expressed as (Ref. 3):

$$I_{sp_{th}} = \overline{F}/\dot{m} \qquad (2.10)$$

where

$$\overline{F} = C_{f_{th}} P_c A_{th} \qquad (2.11)$$

Then

$$A_{th} = \frac{\overline{F}}{C_{f_{th}} \eta C_f P_c} \qquad (2.12)$$

The theoretical throat area can be estimated by using $C_{f_{th}}$ for shifting equilibrium flow (from 'MICROPEP" - the theoretical performance computation). The losses which are calculated to obtain ηC_f depend upon the output from "MICROPEP" and the throat area. Thus, iteration is required to obtain both the losses and throat area.

In summary, the calculation can be conducted in the following steps:

1. Calculate A_{th} using equation (2.11).

2. Calculate η_{C_f} as discussed above.

3. Calculate A_{th} using equation (2.12).

4. Repeat steps 2. and 3. until the values for throat area and η_{C_f} converge.

In the current program the iteration was terminated when there was less than a 1% change in A_{th} per iteration.

The exit area can then be calculated from

$$A_e = A_{th} \, \varepsilon \tag{2.13}$$

where ε is the exit to throat area ratio. ε is calculated in "MICROPEP", assuming that the nozzle exit pressure is equal to the local ambient pressure (one-dimensional, ideal flow is assumed).

H. GRAIN DESIGN

The primary purpose of grain design is to shape the propellant grain such that the rocket motor can deliver the required performance (thrust-time or pressure-time profile).

In order to be able to select the general configuration of the grain geometry, several parameters must first be determined: the web fraction (W_f), volumetric loading fraction (V_ℓ), and the length-to-diameter ratio (L/D) (Ref. 5). L/D is known from the input data. Web fraction is the fraction of the motor radius that is filled with propellant. Thus,

$$W_f = \frac{\dot{r} t_b}{(D/2)} \tag{2.14}$$

The propellant burning rate ($\dot{r}$) is generally expressed in the form:

$$r = ap_c^n \tag{2.15}$$

where a and n are empirical constants. They are assumed to be known for the selected propellant. The volumetric loading fraction is defined as the ratio of the volume occupied by the propellant (V_p) to the total volume available in the motor for propellant (V_a). Thus,

$$V_\ell = \frac{V_p}{V_a} \tag{2.16}$$

where

$$V_a = \left(\frac{\pi D^2}{4}\right) L \tag{2.17}$$

The required propellant volume is found as shown below

$$I_t = \bar{F}t_b = I_{sp}\dot{m}t_b = I_{sp}\,\frac{W_p}{t_b}\,t_b = I_{sp}W_p \tag{2.18}$$

Thus, $W_p = I_t/I_{sp}$ (2.19)

This yields the required propellant weight (W_p) and

$$V_p = \frac{W_p}{\rho_p} \tag{2.20}$$

where ρ_p is the density or specific weight of the selected propellant. With W_f, $V\ell$, and L/D known, an initial selection of grain geometry can be made (for example, see Ref. 5). For a selected grain configuration, the web versus burning surface area (which is the data needed for

thrust and pressure versus time calculations) can be computed.

I. THRUST AND PRESSURE VERSUS TIME CALCULATIONS

The thrust and pressure versus time calculations are made using conservation of mass for the combustor(Ref. 3). The mass addition from the burning propellant ($\dot{m}_g$) is equal to the sum of the mass accumulated in the combustor and the mass exhausted from the nozzle ($\dot{m}_n$) as shown in Figure 4. Thus,

$$\dot{m}_g = dM/dt + \dot{m}_n \tag{2.21}$$

$\dot{m}_g$ is the mass of gas generated from the burning propellant.

$$\dot{m}_g = \sum_i \rho_p A_b \dot{r} = \sum \rho_p A_b a P_c^{\,n} \tag{2.22}$$

The summation is over all propellant grains that are burning at the same time.

$\frac{dM}{dt}$ is the mass accumulating in the combustor.

$$\frac{dM}{dt} = \frac{d}{dt}(\rho_g Vol) = \rho_g \frac{dVol}{dt} + Vol\,\frac{d\rho_g}{dt}$$

where (2.23)

$$\frac{dVol}{dt} = \sum_i \dot{r} A_b \quad \text{and} \quad \frac{d\rho_g}{dt} \cong \frac{1}{RT_c}\,\frac{dP_c}{dt}$$

then

$$\frac{dM}{dt} = \frac{P_c}{\bar{R}T_c}\,\frac{dVol}{dt} + \frac{vol}{\bar{R}T_c}\,\frac{dP_c}{dt} \tag{2.24}$$

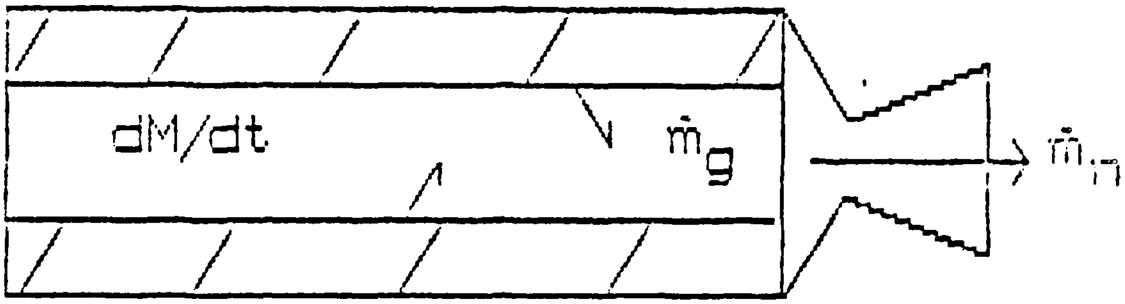


FIGURE 4.

MASS CONSERVATION IN A SOLID PROPELLANT MOTOR

$\dot{m}_n$ is the mass exhausted from the nozzle. For chocked flow,

$$\dot{m}_n = \frac{P_c\ A_{th}\ g_c}{C^*} \tag{2.25}$$

Combining equations 2.21 through 2.25 results in

$$\sum_i \rho_p A_b a P_c^n = \left(\frac{P_c}{\bar{R}T_c}\right)\frac{dVol}{dt} + \left(\frac{Vol}{\bar{R}T_c}\right)\frac{dP_c}{dt} + \left[\frac{P_c A_{th} g_c}{C^*}\right] \tag{2.26}$$

Solving for dP_c/dt

$$\frac{dP_c}{dt} = \left(\frac{1}{Vol}\right)\left[\bar{R}T_c\left(\sum_i \rho_p A_b a P_c^n - \frac{P_c A_{th} g_c}{C^*}\right) - P_c\left(\frac{dVol}{dt}\right)\right] \tag{2.27}$$

Equation 2.27 can be numerically integrated to give $P_c(t)$. The thrust-time behavior can then be obtained from

$$F\ (t) = \eta_{c_f}\ C_f(t) P_c(t) A_{th}(t) \tag{2.28}$$

J. TERMINAL VELOCITY AND RANGE CALCULATIONS

The last step of solid propellant rocket motor design is to incorporate the designed motor into the desired airframe and make a flight simulation. The purpose of this flight simulation is to verify that the motor can produce the desired range and terminal velocity.

III. MICROPEP

"MICROPEP" is a microcomputer version of the propellant Evaluation Program (PEP), published in Naval Weapons Center Report NWC-TP-6037, "Theoretical Computations of Equilibrium Compositions, Thermodynamic Properties, and Performance Characteristics of Propellant Systems," by D. R. Cruise, April 1979 (Ref. 4). The code (MICROPEP) can handle 12 different input ingredients and 200 combustion species. It performs the calculations for the thermodynamic properties and performance of propellant systems.

A. BASIC ASSUMPTIONS

The basic assumptions of "MICROPEP" are as follows:

1. Equilibrium adiabatic combustion at the nozzle entrance.
2. One-dimensional, isentropic expansion in the nozzle in which the flow can be either a shifting equilibrium flow (assuming chemical equilibrium is maintained throughout the expansion process) or frozen composition flow (assuming the chemical composition of the the flow remains the same throughout the expansion process).
3. Gases were considered to be perfect.
4. Nozzle exit pressure is assumed to be equal to the ambient pressure.

B. INPUTS

Inputs required by "MICROPEP' are:

1. Composition and mass of each ingredient. Normally the weights of ingredients are chosen to add up to 100 g.

   NOTE: Ingredients may be input by the user or they may be selected from an existing data file. In the latter case ingredient serial numbers are used.

   For assistance, use README, which should be on the PEPCODED.DAF Data File.

2. Combustion pressure.
3. Nozzle exit pressure. This defines the limits for the expansion process.

C. OUTPUTS

The program calculates:

1. Combustion product composition, properties , and temperature at the nozzle entrance.
2. Nozzle exit product composition, properties, and temperature for two cases:
   a. Shifting equilibrium flow
   b. Frozen composition flow.
3. Required throat area and exit area with no losses, for the same two cases.
4. Theoretical specific impulse, thrust coefficient, and characteristic exhaust velocity for both shifting equilibrium and frozen composition flow.

IV. LOSS MECHANISMS

For estimation of the performance losses, a set of empirical formulas suggested by the Propulsion and Energetics Panel of AGARD in AGARD-AR-230 (Ref. 6) were used. Similar methods can also be found in NASA Report SP-8039 (Ref. 7) "Solid Rocket Motor Performance Analysis and Prediction." It is assumed, as in equation 2.7, that the combustion efficiency (η_{C^*}) and thrust coefficient efficiency (η_{C_f}) can be treated independently. The empirical formulas are valid only for aluminized propellants.

A. THRUST COEFFICIENT EFFICIENCY

The thrust coefficient efficiency (η_{C_f}) can be expressed as in equation 2.8

$$\eta_{C_f} = 1-(\varepsilon_{DIV}+\varepsilon_{TP}+\varepsilon_{BL}+\varepsilon_{KIN}+\varepsilon_{SUB}+\varepsilon_{EROS})/100.0$$

Empirical equations for each of the losses (ε's) will be presented and briefly discussed below. More details can be found in Reference 6.

1. Divergence Loss (ε_{DIV})

This is the loss due to the radial velocity component of the gas at the nozzle exit plane

$$\varepsilon_{DIV} = 50\ [1-\mathrm{Cos}\ (\frac{\alpha+\theta_{ex}}{2})] \qquad (4.1)$$

α is the nozzle half angle and θex is the exit angle for a contoured nozzle. For a contoured nozzle, α is measured to a line drawn from the exit and tangent to the wall at the throat.

2. Kinetics Loss (ε_{KIN})

This is the loss (or reduction in performance from the value of shifting equilibrium flow) associated with not attaining chemical equilibrium throughout the nozzle expansion process. It is estimated to be 1/3 of the fractional difference between the specific impulse for shifting equilibrium (I_{sp_s}) and frozen composition (I_{sp_f}). Thus,

$$\varepsilon_{KIN} = 33.3 \left[1 - \frac{I_{sp_f}}{I_{sp_s}}\right] \tag{4.2}$$

Both I_{sp_s} and I_{sp_f} are obtained from 'MICROPEP."

3. Boundary Layer Loss (ε_{BL})

The boundary layer loss (ε_{BL}) is expressed as

$$\varepsilon_{BL} = C_1 \frac{P_c^{0.8}}{D_{th}^{0.2}} \left[1+2 \exp\left(-C_2 \frac{P_c^{0.8}}{D_{th}^{0.2}} t_b\right)\right] \left[1+0.016(\varepsilon-9)\right] \tag{4.3}$$

This loss accounts for both reduced flow areas and transient heat loss. In equation 4.3, P_c is in psi, D_{th} is in inches. The C_1 and C_2 coefficients are

for ordinary nozzles $C_1 = 0.00365$

$C_2 = 0.000937$

for steel nozzles $C_1 = 0.00506$

$C_2 = 0.0$

4. Two-Phase Flow Loss (ε_{TP})

This loss is due primarily to the velocity lag of the condensed species of the combustion products as the mixture passes through the exhaust nozzle,

$$\varepsilon_{TP} = C_3 \frac{M_f^{C_4} \quad D_p^{C_5}}{P_c^{0.15} \quad \varepsilon^{0.08} \quad D_{th}^{C_6}} \tag{4.4}$$

where

a. D_{th} is the diameter of throat in inches

b. D_p is the mean Al_2O_3 particle diameter in microns and is calculated from

$$D_p = 3.39 \; D_{th}^{0.4692}$$

c. P_c is chamber pressure in psia

d. C's are dependent on the diameter of the throat

$C_4 = 0.5$

$D_{th} < 1$:	$C_3 = 9$	$C_5 = 1$	$C_6 = 1$
$1 \leq D_{th} \leq 2$:	$C_3 = 9$	$C_5 = 1$	$C_6 = 0.08$
$D_{th} > 2$ and $D_p < 4$:	$C_3 = 13.4$	$C_5 = 0.8$	$C_6 = 0.8$
$D_{th} > 2$ and $4 \leq D_p \leq 8$:	$C_3 = 10.2$	$C_5 = 0.8$	$C_6 = 0.4$
$D_{th} > 2$ and $D_p > 8$:	$C_3 = 7.58$	$C_5 = 0.8$	$C_6 = 0.33$

e. M_f is the mole fraction of consdensed phase in moles/100 gm of reactants.

5. <u>Submergence Loss (ε_{SUB})</u>

This loss is expressed as

$$\varepsilon_{SUB} = 0.0684 \left(\frac{P_c \varepsilon}{A_{th}}\right)^{0.8} \frac{S^{0.4}}{D_{th}^{0.2}} \qquad (4.5)$$

It is due to the nozzle configuration for a submerged nozzle where S is the submergence length in inches (the length that the nozzle is imbedded in the combustion chamber).

6. <u>Nozzle Erosion Loss (ε_{EROS})</u>

This loss results from nozzle erosion during the burn and is expressed as

$$\varepsilon_{EROS} = \left[1 - \frac{I_{sp_m}}{I_{sp_{th}}}\right] \times 100 \qquad (4.6)$$

$I_{sp_{th}}$ is the theoretical I_{sp} for the initial nozzle expansion ratio and I_{sp_m} is the theoretical I_{sp} for the mean expansion ratio. I_{sp_m} is determined by interpolation of $I_{sp_{th}}$ versus the nozzle expansion ratio data from "MICROPEP."

B. COMBUSTION EFFICIENCY

NASA Report SP-8064 (Ref. 2) states that the combustion efficienty (η_{C^*}) is determined by "the completeness of metal combustion within the motor and by the degree to which combustion products reach chemical equilibrium among themselves." These effects depend primarily on the residence time (t_r) as expressed in equation 2.9. An η_{C^*} of

90-95% can be attained if residence time is 10-15 msec (see Figure 3).

C. PROGRAM

The subroutines which solve for the thrust and combustion efficiencies are listed in Appendix A, Subroutine PFE and Subroutine LOSCF. Basically, they followed the method discussed in Chapter II. The algorithm is shown below

1. Initialize C_f (coefficient of thrust) as $C_{f_{th}}$ (theoretical coefficient of thrust from "MICROPEP" for shifting equilibrium).

2. Calculate A_{th} and D_{th}:

$$D_{th} = 2\left(\frac{\bar{F}}{C_{f_{th}} P_c \pi}\right)^{1/2}$$

3. Calculate η_{c_f} using the relation

$$\eta_{c_f} = 1-(\varepsilon_{DIV}+\varepsilon_{BL}+\varepsilon_{TP}+\varepsilon_{KIN}+\varepsilon_{SUB}+\varepsilon_{EROS})/100$$

4. Calculate C_f using

$$C_f = C_{f_{th}} \eta_{c_f}$$

5. Correct D_{th} using

$$D_{th} = 2\left(\frac{\bar{F}}{C_f \cdot P_c \pi}\right)^{1/2}$$

6. Repeat Steps 3 to 5 until A_{th} converges.

The reason that A_{th} was used for the iteration variable was that η_c (through ε_{BL}, ε_{TP}, and ε_{SUB}) depends upon A_{th}.

V. GRAINS

"GRAINS" iS PC based code for the design of radial sided star grains. The original version was written by G. J. Woten and R. McCormick of CSD, United Technologies, November 1984. It was then modified by LT G. Liston of AFWAL/Port, December 1984. The general shape of this grain and the geometric variables are shown in Figure 5.

"GRAINS" can be used in either of two modes: the "Design Mode" or the "Burnback Model."

A. "DESIGN MODE"

In the "Design mode" the program solves for geometries that satisfy input requirements on the number of star points required port area, perimeter factor, web, and etc., together with the allowed tolerances on these parameters.

The geometric constraints for the grain are determined from the propellant characteristic and motor constraints. For example:

$$\text{Aport} = \left(\frac{\pi}{4}\right)D^2\left[\left(1 - \left(\frac{\overline{F}t_b}{I_{sp}\rho_p}\right)\right]\right.$$

D, t_b and $\overline{F}$ are motor inputs requirements and I_{sp} and ρ_p are determined once the propellant has been selected.

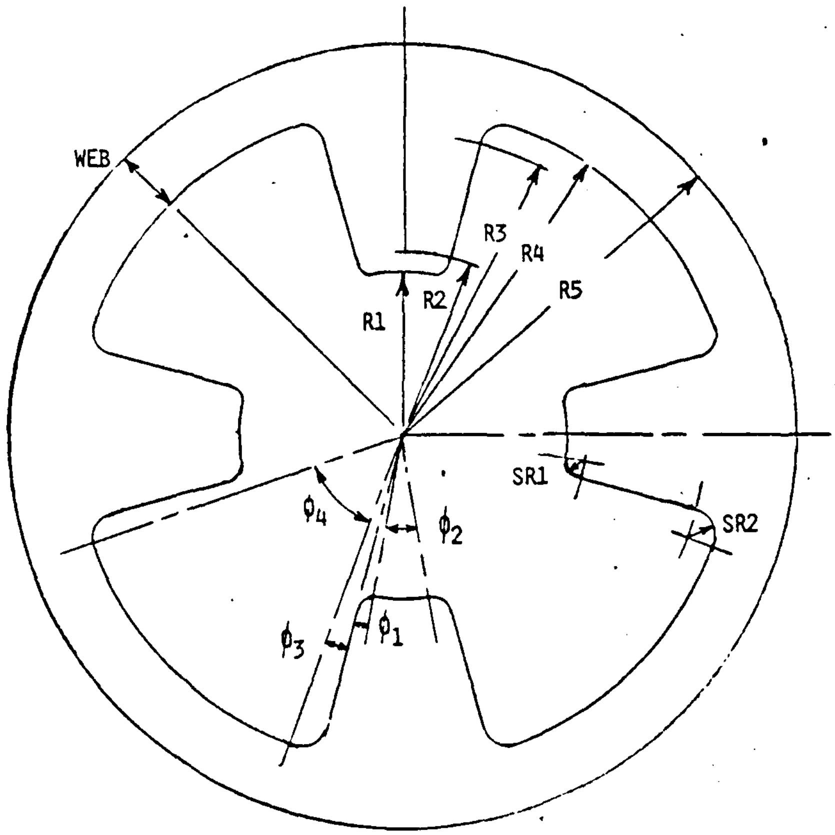


FIGURE 5.

NOMENCLATURE FOR "GRAINS" DESIGN MODE

B. "BURNBACK MODE"

The "burnback mode" uses the length of the grain, initial port area, perimeter, and the results from "Design Mode" to calculate the perimeter, perimeter factor, port area, and burning surface area as functions of the web burned.

C. COMMENTS ON PROGRAM OPERATION

Care must be used for the inputs and input tolerances, otherwise no acceptable solutions will be obtained. It is best to start with large allowable tolerances on A_{port}, etc. (typically 20%). After initial solutions are found in "Design" for the number of star points, range of web thicknesses, etc., the range of the input variables can be reduced along with the allowable tolerances.

VI. ROCKET

"ROCKET" is also a PC-based computer code. It was originally written by The Lockheed Company for a mainframe computer. Later, it was transferred for use on the IBM-PC by J. P. Francis of the Naval Weapons Center (August 1985). The function of "ROCKET" is to calculate the pressure-time and thrust-time profiles for the motor, including theoretical calculations for both ignition and tailoff transients. The description of the motor can include igniters, multiple propellant grains, and motor insulation. In this thesis work the program was restricted to a motor with a single grain configuration.

A. BASIC ASSUMPTIONS

The assumptions made in "ROCKET" are:

1. The combustion products are ideal gases.
2. Propellant burning rate follows the expression $\dot{r} = aP_c^n$.
3. There are no effects of mass addition or erosive burning in the combustion chamber.
4. C* varies with P_c in the form of

$$C^* = C^*_{ref}\left(\frac{P_c}{P_{ref}}\right)^x$$

5. The temperature sensitivity (π_K) is used to correct for the effects of propellant temperature on the propellant burning rate and chamber pressure.

$$\dot{r}_{T_p} = \dot{r}_{(70°)}\, e^{0.01\sigma_p(T_p - 70°)}$$

where

$$\sigma_p = (1-n)\pi_K$$

B. ANALYTICAL BACKGROUND

The program is based upon the method discussed in Chapter II, i.e., it numerically integrates equation (2.27) to obtain $P_c(t)$. Then it calculates C_f with a fixed ratio of specific heats (γ). Thrust is then calculated using equation

$$F = C_f(t)P_c(t)A_{th}(t)\lambda C_D$$

where λ is nozzle divergence loss and C_D is the nozzle discharge coefficient. λ and C_D are only two parts of η_{C_f}. In order to use ROCKET in the combined program, λ was set to unity and C_D was equated to η_{C_f}.

C. INPUTS

Inputs to "ROCKET" include:

1. General motor parameters:
   a. Throat area (A_{th})
   b. Nozzle exit area (A_e)
   c. Total motor volume (V_a)
   d. Ambient temperature and pressure

e. Nozzle throat plug closure blowout pressure (nozzle is plugged until this pressure is exceeded, typically 35 psia)

f. Initial pressure in the motor (P_{zero}, typically 15 psia)

g. Nozzle divergence loss λ (set equal to 1 in this work)

h. Ratio of specific heats for chamber gases (γ) (this was obtained from "MICROPEP" chamber conditions)

i. Design pressure (P_c)

j. Throat radial erosion rate (erosr).

2. Motor element descriptions:

a. Propellant burning rate ($\dot{r}$)

b. Burning rate exponent (n)

c. Propellant temperature sensitive (π_K)

d. Burning rate reference pressure

e. Propellant C*

f. Pressure correction exponent for C*

g. Propellant density (ρ_p)

h. A table of web versus burning surface area.

D. OUTPUTS

The outputs include:

1. A pressure versus time table

2. A thrust versus time table

3. The free volume, propellant consumed (weight), and throat area versus time table.

VII. FLYIT

"FLYIT" is a three-degree-of-freedom (range, altitude and angle of attack) trajectory simulation program originally developed by G. Burgner of the Propulsion Analysis Branch, Naval Weapons Center (NWC). A User's guide was later published by Y.G. Coenen (Ref. 8). The program can simulate vertical-plane trajectories which include take-off, climbs, cruises, landing, and rocket boost. It is a tool to synthesize and analyze the trajectories of air-to-surface, and surface-to-surface missiles.

The basic structure of "FLYIT" consists of five functional sub-groups, they are:

- Atmospherics
- Trajectory controls
- Aerodynamics
- Rocket boosters
- Sustanier propulsion.

Users are allowed to select a different option in each functional group to simulate different flight profiles with different propulsion systems. Through the use of "FLYIT" it is possible to estimate the terminal velocity and range of certain propulsion systems.

"FLYIT" was not incorporated into the main body of the present program due to time limitations.

VIII. COMBINED PROGRAM

The combined program ("SPRMD") is listed in Appendix and basically follows the design process discussed in Chapter II. The basic structure of this program is shown on Figure 6.

This program combines "MICROPEP" (to calculate the theoretical thermodynamic performance of the propellant), "PFE," and "LOSCF" (to calculate η_{C_f}, and to estimate η_{C^*}, and size the throat and exit areas), "GRAINS" (to provide web burned versus burning surface area for radially sided star grains), and "ROCKET" (to compute the pressure-time and thrust-time profiles). Inputs to the program include "INPUT0" (the initial mission requirement inputs), "INPUT1" (inputs for "MICROPEP"), "INPUT2" (inputs for "GRAINS"), and "RCKTIN" (inputs for "ROCKET"). All of these programs comprise an interactive program for solid propellant rocket motor preliminary design.

A. ALGORITHM

A simplified algorithm of this program can be briefly listed as follows:

1. "INPUT0"

   Input the basic ballistic and mission/vehicle requirements. Calculate the required average

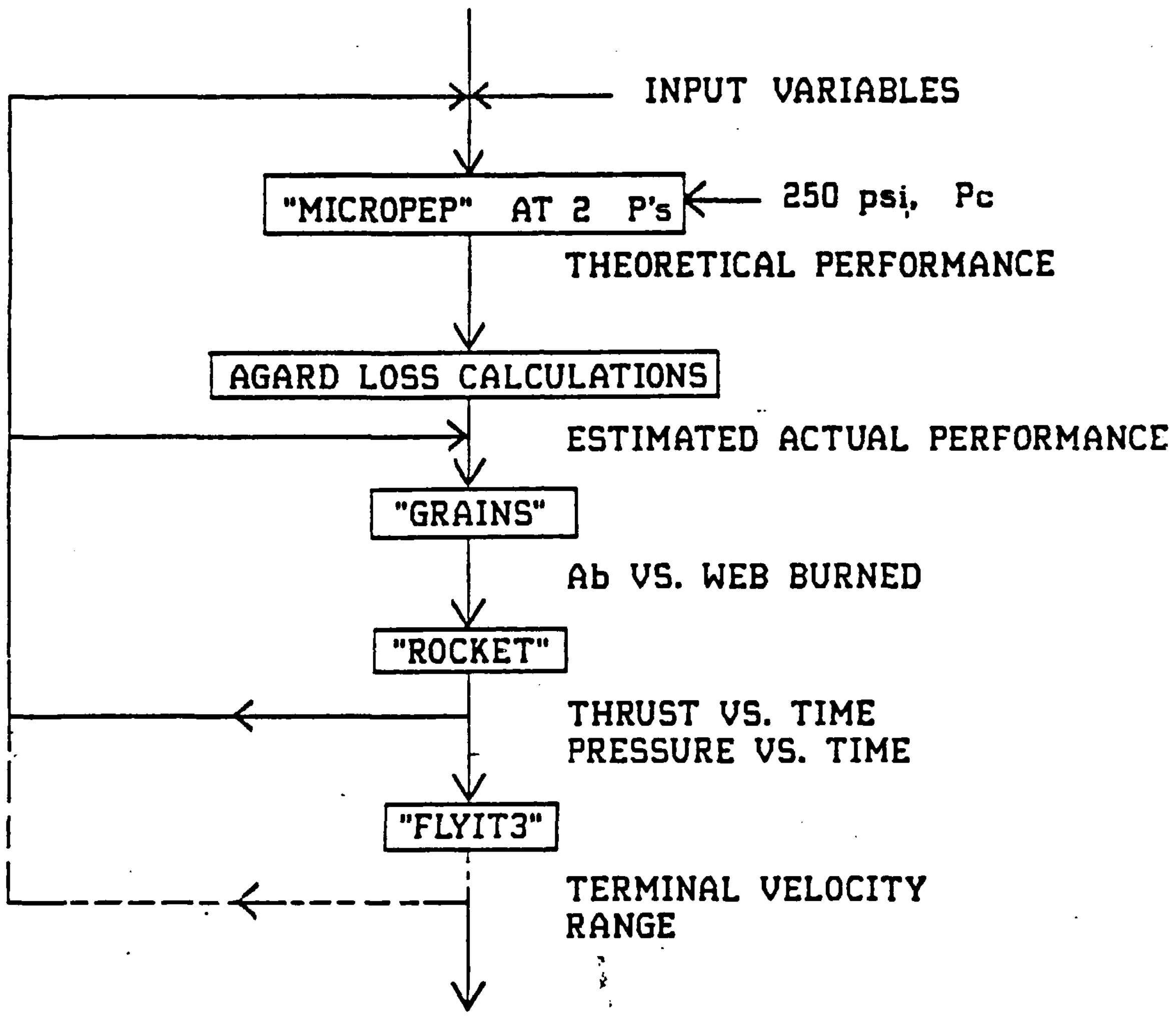


FIGURE 6.

FLOW CHART FOR THE COMBINED PROGRAM

requirements. Calculate the required average chamber pressure for a specific propellant and propellant temperature.

2. "INPUT1"

Input propellant composition, chamber pressure, and ambient pressure. Create a file for 'MICROPEP."

3. "MICROPEP"

Perform the theoretical performance calculations for the selected propellant.

4. "PFE"

Performs the following functions:

a. Estimates the combustion efficiency

b. Calculates the delivered specific impulse

c. Sizes the throat area, exist area, calculates perimeter factor, perimeter, required web, required burning rate, and etc.

5. "INPUT2"

Input the geometrical description of the star grain and propellant characteristics. Create a file to accommodate the processing of "GRAINS." Other grain design programs (such as SPP) can be readily incorporated.

6. "GRAINS"

Designs the star grain and calculates web versus burning surface area.

7. "RCKTIN"

Uses the results from the previous programs to create a file for "ROCKET."

8. "ROCKET"

Calculates the pressure-time and thrust-time profiles.

B. BRANCHING

The algorithm discussed above also provides for the capability of branching. For example, if the desired P_c versus time is not attained, the user can either re-do the grain design and/or select other propellant properties.

IX. EXAMPLE OF DESIGN PROGRAM

The initial requirements which are input into the program are shown in Table 1, together with a selected value for the temperature sensitivity of the propellant (π_K). A reproducibility tolerance of 7% was chosen.

The nominal operating pressure at T_p = 70°F was calculated to be 1030 psia and the required propellant burning rate for a web fraction of 0.5 was 0.615 in/sec.

A composite propellant consisting of 78% AP, 12% HTPB, and 10% Al was selected. "MICROPEP" was then run for pressures of 250 psia and the nominal P_c value of 1030 psi. Outputs from "MICROPEP" are presented in Table 2.

TABLE 1

INPUT PARAMETERS

AVERAGE THRUST, LBF....................	14250.0
BURN TIME, SEC.........................	6.50
MINIMUM THRUST REQUIRED, LBF..........	8000.000000
MOTOR LENGTH, INCH.....................	40.00
MOTOR DIAMETER, INCH...................	16.00
MAXIMUM EXPECTED OPERATING PRESSURE...	1300.0
TPMAX, F...............................	150.0
TPNOM, F...............................	70.0
TPMIN, F...............................	-65.0
TEMPERATURE SENSITIVITY, %/F..........	0.200
NOMINAL CHAMBER PRESSURE, PSI........	1030.2
REQUIRED BURNING RATE, IN/SEC.........	0.615
ALTITUDE, FT...........................	10000.
AMBIENT PRESSURE, PSI..................	10.13
AMBIENT TEMPERATURE, R.................	483.1

TABLE 2

MICROPEP OUTPUT FOR P_C = 250 AND 1030 PSIA: P_C = 10.0 PSIA. 78% AP, 12% HTPB, 10% Al

```
DESIGN 1    Case   1 of   2

                                            DH    DENS COMPOSITION

AMMONIUM PERCHLORATE (AP)                 -602  0.07040    1CL  4H    1N   4O
HTPB (SINCLAIR)                             13  0.03320  103H  73C   1O
ALUMINUM (PURE CRYSTALINE)                   0  0.09760    1AL

INGREDIENT WEIGHTS (IN ORDER) AND TOTAL WEIGHT        (LAST ITEM IN LIST)

   78.0000     12.0000     10.0000    100.0000

INGREDIENT VOLUME RATIOS
AMMONIUM PERCHLORATE (AP)       70.487%   HTPB (SINCLAIR)                  22.995%
ALUMINUM (PURE CRYSTALINE)       6.518%

THE PROPELLENT DENSITY IS 0.06362 LB/CU-IN OR  1.7610 GM/CC

NUMBER OF GRAM ATOMS OF EACH ELEMENT PRESENT IN INGREDIENTS

   3.895570 H      0.878965 C      0.663847 N      2.667428 O
   0.370645 AL     0.663847 CL
 CP =       43.62668610

CHAMBER RESULTS FOLLOW:

 T(K)  T(F)  P(ATM)    P(PSI) ENTHALPY  ENTROPY   CP/CV    GAS     RT/V    MOL WT
3248. 5388.   17.01    250.00   -46.80   240.56  1.1990  3.644    4.667   27.44

DAMPED AND UNDAMPED SPEED OF SOUND= 3211.743 AND 3563.987 FT/SEC

HEAT CONTENT (298 REF)  1253.397 CAL/GR       2257.631 BTU/LBM

  0.98321 H2O         0.71991 CO          0.58380 H2          0.57339 HCl
  0.32835 N2          0.17947 Al2O3*      0.15900 CO2         0.11256 H
  0.07546 HO          0.07476 Cl          0.00880 O           0.00709 NO
 5.93E-03 O2         4.51E-03 AlOCl      3.65E-03 AlCl2      3.20E-03 AlCl
 1.32E-04 AlCl3      1.28E-04 Cl2        1.19E-04 AlO        6.37E-05 Al
 2.31E-05 CHO        1.73E-05 NH         1.61E-05 N          1.21E-05 HO2
 7.13E-06 NH3        5.55E-06 NHO        5.24E-06 Al2O       4.85E-06 AlH
 2.03E-06 CNH        1.91E-06 AlHO2      1.09E-06 CH2O
 CP =       41.92294693

EXHAUST RESULTS FOLLOW:

 T(K)  T(F)  P(ATM)    P(PSI) ENTHALPY  ENTROPY   CP/CV    GAS     RT/V    MOL WT
2313. 3704.    0.69     10.13  -108.91   240.56  1.1995  3.510    0.196   28.49

DAMPED AND UNDAMPED SPEED OF SOUND= 2650.051 AND 2952.060 FT/SEC

HEAT CONTENT (298 REF)   851.166 CAL/GR       1533.130 BTU/LBM

  1.01958 H2O         0.67108 CO          0.64916 HCl         0.59218 H2
  0.33181 N2          0.20785 CO2         0.18284 Al2O3&      0.01785 H
  0.01421 Cl          0.00497 HO          0.00233 Al2O3*      0.00021 NO
 1.33E-04 AlCl2      1.04E-04 O          1.01E-04 AlOCl      7.51E-05 O2
 4.03E-05 AlCl       1.25E-05 AlCl3      9.06E-06 Cl2
```

TABLE 2 CONTINUED)

MICROPEP OUTPUT FOR P_C = 250 AND 1030 PSIA:
P_C = 10.0 PSIA. 78% AP, 12% HTPB, 10% Al

PERFORMANCE: FROZEN ON FIRST LINE, SHIFTING ON SECOND LINE

ISP	GAMMA	T*	P*	C*	CF	OPT EX	D-ISP	A*M.	EX T
226.7	1.1835	2975.	9.65	5060.1	1.442	4.31	399.2	0.62924	1976.
232.5	1.1497	3045.	9.77	5082.8	1.472	4.72	409.4	0.63205	2313.

BOOST VELOCITIES FOR PROPELLANT DENSITY OF 0.06362 (S.G. OF 1.761)

5./23451.	10./18585.	15./15857.	25./12612.	30./11520.	55./ 8215.
60./ 7788.	69./ 7128.	71./ 6998.	88./ 6064.	100./ 5548.	150./ 4113.
175./3[illegible]7.	200./3277.	300./2336.	1000./ 780.	3000./ 269.	5000./ 163.

EXP. RATIO	EXIT PRESS ATM	EXIT PRESS SI	EXIT TEMP K	OPTIMUM IMPULSE SEC	OPTIMUM IMPULSE SI	VACUUM IMPULS SEC	VACUUM IMPULS SI	SEA LV IMPULS SEC	SEA LV IMPULS SI
1.	9.769	989.6	3044.	104.9	1029.	195.7	1919.	186.4	1846.
2.	3.256	329.9	2678.	175.0	1716.	235.5	2309.	216.9	2149.
3.	1.215	123.0	2386.	214.4	2103.	248.3	2435.	220.4	2184.
4.	0.833	84.4	2314.	226.6	2223.	257.6	2526.	220.5	2184.
5.	0.644	65.3	2313.	234.5	2300.	264.5	2593.	218.0	2160.
6.	0.523	53.0	2311.	240.7	2361.	269.9	2647.	214.1	2121.
7.	0.439	44.5	2310.	245.8	2410.	274.4	2690.	209.3	2074.
8.	0.378	38.3	2310.	250.1	2453.	278.2	2728.	203.9	2019.
9.	0.331	33.5	2309.	253.8	2489.	281.5	2760.	197.9	1960.
10.	0.294	29.8	2308.	257.1	2521.	284.4	2789.	191.5	1897.
11.	0.264	26.8	2307.	260.0	2550.	287.0	2814.	184.8	1831.
12.	0.240	24.3	2307.	262.6	2575.	289.3	2837.	177.9	1762.
13.	0.219	22.2	2306.	265.0	2599.	291.5	2858.	170.7	1691.
14.	0.202	20.5	2306.	267.2	2620.	293.5	2878.	163.4	1619.
15.	0.187	19.0	2305.	269.2	2640.	295.3	2896.	155.9	1545.
16.	0.174	17.7	2305.	271.1	2658.	297.0	2912.	148.3	1469.
17.	0.163	16.5	2305.	272.8	2675.	298.5	2928.	140.6	1393.
18.	0.153	15.5	2304.	274.4	2691.	300.0	2942.	132.8	1316.
19.	0.144	14.6	2304.	276.0	2706.	301.4	2956.	124.9	1237.
20.	0.136	13.8	2304.	277.4	2720.	302.7	2969.	116.9	1158.
21.	0.129	13.1	2303.	278.8	2734.	304.0	2981.	108.9	1079.
22.	0.123	12.4	2303.	280.1	2746.	305.1	2992.	100.8	998.
23.	0.117	11.8	2303.	281.3	2758.	306.3	3003.	92.6	917.
24.	0.112	11.3	2302.	282.5	2770.	307.3	3014.	84.4	836.
25.	0.107	10.8	2302.	283.6	2781.	308.3	3024.	76.1	754.

TABLE 2 (CONTINUED)

MICROPEP OUTPUT FOR P_C = 250 AND 1030 PSIA: P_C = 10.0 PSIA. 78% AP, 12% HTPB, 10% Al

DESIGN 1 Case 2 of 2

	DH	DENS	COMPOSITION			
AMMONIUM PERCHLORATE (AP)	-602	0.07040	1CL	4H	1N	4O
HTPB (SINCLAIR)	13	0.03320	103H	73C	1O	
ALUMINUM (PURE CRYSTALINE)	0	0.09760	1AL			

INGREDIENT WEIGHTS (IN ORDER) AND TOTAL WEIGHT (LAST ITEM IN LIST)

78.0000 12.0000 10.0000 100.0000

INGREDIENT VOLUME RATIOS
AMMONIUM PERCHLORATE (AP) 70.487% HTPB (SINCLAIR) 22.995%
ALUMINUM (PURE CRYSTALINE) 6.518%

THE PROPELLENT DENSITY IS 0.06362 LB/CU-IN OR 1.7610 GM/CC

NUMBER OF GRAM ATOMS OF EACH ELEMENT PRESENT IN INGREDIENTS

3.895570 H 0.878965 C 0.663847 N 2.667428 O
0.370645 AL 0.663847 CL
CP = 43.75085831

CHAMBER RESULTS FOLLOW:

T(K)	T(F)	P(ATM)	P(PSI)	ENTHALPY	ENTROPY	CP/CV	GAS	RT/V	MOL WT
3375.	5616.	70.08	1030.24	-46.80	230.37	1.1952	3.597	19.485	27.80

DAMPED AND UNDAMPED SPEED OF SOUND= 3247.693 AND 3603.495 FT/SEC

HEAT CONTENT (298 REF) 1309.442 CAL/GR 2358.580 BTU/LBM

1.01084 H2O	0.71716 CO	0.59375 HCl	0.57367 H2
0.32895 N2	0.17941 Al2O3*	0.16171 CO2	0.07526 H
0.05727 HO	0.05233 Cl	0.00586 NO	0.00506 AlCl2
4.49E-03 O	3.70E-03 AlOCl	3.15E-03 O2	2.63E-03 AlCl
3.14E-04 AlCl3	1.83E-04 Cl2	7.24E-05 AlO	5.77E-05 CHO
3.63E-05 Al	2.75E-05 NH	2.71E-05 NH3	1.55E-05 N
1.33E-05 HO2	9.39E-06 NHO	8.06E-06 CNH	5.08E-06 AlH
4.45E-06 CH2O	3.50E-06 Al2O	2.16E-06 AlHO2	

CP = 40.18283081

EXHAUST RESULTS FOLLOW:

T(K)	T(F)	P(ATM)	P(PSI)	ENTHALPY	ENTROPY	CP/CV	GAS	RT/V	MOL WT
1852.	2874.	0.69	10.13	-130.14	230.37	1.2087	3.491	0.197	28.64

DAMPED AND UNDAMPED SPEED OF SOUND= 2373.392 AND 2644.581 FT/SEC

HEAT CONTENT (298 REF) 659.260 CAL/GR 1187.465 BTU/LBM

0.98547 H2O	0.66303 HCl	0.63198 CO	0.63022 H2
0.33192 N2	0.24696 CO2	0.18531 Al2O3&	0.00097 H
7.87E-04 Cl	1.10E-04 HO	2.78E-06 AlCl2	2.25E-06 NO
2.21E-06 AlCl3	1.52E-06 NH3		

PERFORMANCE: FROZEN ON FIRST LINE, SHIFTING ON SECOND LINE

TABLE 2 (CONTINUED)

MICROPEP OUTPUT FOR P_C = 250 AND 1030 PSIA:
P_C = 10.0 PSIA. 78% AP, 12% HTPB, 10% Al

ISP	GAMMA	T*	P*	C*	CF	OPT EX	D-ISP	A*M.	EX T
262.5	1.1907	3081.	39.69	5084.9	1.661	12.28	462.2	0.15344	1610.
269.3	1.1617	3139.	40.09	5126.9	1.690	13.25	474.2	0.15471	1852.

BOOST VELOCITIES FOR PROPELLANT DENSITY OF 0.06362 (S.G. OF 1.761)

5./27165.	10./21528.	15./18368.	25./14609.	30./13344.	55./ 9516.
60./ 9021.	69./ 8257.	71./ 8106.	88./ 7024.	100./ 6426.	150./ 4764.
175./4224.	200./3796.	300./2705.	1000./ 904.	3000./ 312.	5000./ 188.

EXP. RATIO	EXIT PRESS ATM	EXIT PRESS SI	EXIT TEMP K	OPTIMUM IMPULSE SEC	OPTIMUM IMPULSE SI	VACUUM IMPULS SEC	VACUUM IMPULS SI	SEA LV IMPULS SEC	SEA LV IMPULS SI
1.	40.059	4057.9	3139.	106.6	1045.	197.7	1939.	195.4	1936.
2.	13.353	1352.6	2722.	176.8	1734.	237.6	2330.	233.0	2308.
3.	4.904	496.8	2391.	216.7	2125.	250.1	2453.	243.3	2410.
4.	3.289	333.2	2270.	229.5	2250.	259.4	2544.	250.3	2479.
5.	2.435	246.7	2183.	238.3	2337.	266.0	2609.	254.7	2523.
6.	1.913	193.8	2115.	245.0	2402.	271.1	2658.	257.4	2550.
7.	1.563	158.3	2060.	250.3	2454.	275.1	2698.	259.2	2568.
8.	1.314	133.1	2014.	254.6	2497.	278.5	2731.	260.3	2579.
9.	1.129	114.3	1975.	258.3	2532.	281.4	2759.	260.9	2584.
10.	0.986	99.9	1940.	261.4	2563.	283.8	2783.	261.1	2586.
11.	0.873	88.4	1910.	264.2	2590.	286.0	2805.	261.0	2585.
12.	0.781	79.1	1882.	266.6	2615.	287.9	2824.	260.7	2582.
13.	0.706	71.5	1858.	268.8	2636.	289.7	2841.	260.1	2577.
14.	0.643	65.1	1835.	270.8	2655.	291.3	2856.	259.4	2570.
15.	0.589	59.7	1814.	272.6	2673.	292.7	2870.	258.6	2562.
16.	0.543	55.0	1795.	274.3	2690.	294.0	2883.	257.6	2552.
17.	0.503	51.0	1778.	275.8	2705.	295.2	2895.	256.6	2542.
18.	0.468	47.4	1761.	277.2	2718.	296.4	2906.	255.5	2531.
19.	0.438	44.3	1746.	278.5	2731.	297.4	2917.	254.2	2519.
20.	0.411	41.6	1731.	279.8	2743.	298.4	2927.	253.0	2506.
21.	0.386	39.1	1717.	280.9	2755.	299.4	2936.	251.6	2493.
22.	0.365	36.9	1704.	282.0	2766.	300.3	2944.	250.2	2479.
23.	0.345	34.9	1692.	283.0	2776.	301.1	2952.	248.8	2465.
24.	0.327	33.1	1681.	284.0	2785.	301.9	2960.	247.3	2450.
25.	0.311	31.5	1670.	284.9	2794.	302.6	2968.	245.8	2435.

The output from "MICROPEP" indicated the following

C^*_{th} = 5126.9 ft/sec

$I_{sp_{th}}$ = 269.3 lbf sec/lbm

ε = 13.25

Moles of condensed Al_2O_3/100 gm = 0.1794

ρ_p = .0636 lbm/in^3

The AGARD-AR-230 empirical equations were then used to estimate the value for η_{C_f}. η_{C^*} was chosen to be 0.93. For the calculated residence time of 7.2 msec, Figure 3, indicates that a better value may have been approximately 0.87. As discussed above, the losses are iterated with the throat area. The resulting losses and throat area are presented in Table 3.

Once the losses have been estimated I_{sp} can be determined from

$$I_{sp} = \eta_{C_f} \eta_{C^*} I_{sp_{th}}$$

The following parameters were then calculated:

$$\dot{m} = \overline{F}/I_{sp}$$

$$V_p = I_t/I_{sp}\rho_p$$

$$V_\ell = V_p/V_a$$

$$D_{pe} = (\sqrt{1-V_\ell})D$$

$$A_p = (\frac{\pi}{4})D_{pe}^2 = \frac{\pi}{4}\, D^2\, (1-V_\ell)$$

$$J = A_{th}/A_p$$

These parameters are shown in Table 4.

TABLE 3

ESTIMATED PERFORMANCE LOSSES

PERCENT DIVERGENCE LOSS..................	1.704
PERCENT TWO PHASE FLOW LOSS...............	3.202
PERCENT BOUNDARY LAYER LOSS...............	1.828
PERCENT KINETIC LOSS......................	0.849
PERCENT SUBMERGENCE LOSS..................	0.000
PERCENT EROSION LOSS......................	0.007
THRUST COEFFICIENT EFFICIENCY.............	0.924
Etacf= 0.924 Cfc= 1.562 Ath=	8.856
PERCENT DIVERGENCE LOSS..................	1.704
PERCENT TWO PHASE FLOW LOSS...............	3.199
PERCENT BOUNDARY LAYER LOSS...............	1.783
PERCENT KINETIC LOSS......................	0.849
PERCENT SUBMERGENCE LOSS..................	0.000
PERCENT EROSION LOSS......................	0.007
THRUST COEFFICIENT EFFICIENCY.............	0.925

TABLE 4

CALCULATED MOTOR DATA

THRUST COEFFICIENT......................	1.563
ETACF...................................	0.925
SPECIFIC IMPULSE, LBF SEC/LBM.........	231.587
MASS FLOW RATE, LBM/SEC.............	61.532
PROPELLANT VOLUME, CU IN..............	6286.8
NOZZLE THROAT AREA, SQ IN..............	8.851
NOZZLE EXIT AREA, SQ IN..............	117.290
EQUIVALENT PORT DIAMETER, IN...........	7.476
PORT AREA, SQ IN	43.893
PERIMETER FACTOR.......................	2.489
REQUIRED WEB, IN.......................	2.500
REQUIRED WEB FRACTION..................	0.312
VOLUMETRIC LOADING.....................	0.782
LENGTH TO DIAMETER RATIO...............	2.500
RESIDENCE TIME, MSEC...................	7.256
THROAT TO PORT AREA RATIO (J FACTOR)...	0.202
REQUIRED BURNING RATE, IN/SEC........	0.385

At this point the following expressions remain to be satisfied:

$$\left.\begin{array}{l} \text{Perimeter Factor} = PFAC = P_{wetted}/\pi D_{pe} \\ \\ A_b = P_{wetted}\, L \end{array}\right\} PFAC = A_b/\pi D_{pe} L$$

$$\dot{m} = \rho_p\, A_b\, \dot{r}_{P_{cnom}}$$

$$\dot{r} = web/t_b$$

In these three expressions the unspecified variables are PFAC, A_b, $\dot{r}$, and web.

For a selected web or $\dot{r}$ the other parameters can be determined. For example, in the current problem Table E summarizes the required values for selected web thicknesses.

TABLE 5

REQUIRED PARAMETERS FOR SELECTED WEBS

web(in)	$\dot{r}$ (in/sec)	A_b (in^2)	PFAC
3.5	.538	1796	1.91
3.0	.461	2095	2.22
2.5	.385	2514	2.67

Once the required burning rate ($\dot{r}$) is determined, appropriate values of a and n are required for the expression $\dot{r} = aP_c^n$.

The grain design problem is to find a geometry which will satisfy Table 5, together with the required conditions on L, D, P_c-t and F-t (and stress).

In the present program, only radially spoked star grains can be analyzed. It may not be possible to satisfy the motor requirements with this type of grain. The process is essentially one of trial-and-error.

Three attempts at meeting the design requirements are presented. One 6-point and two 8-point designs were tried.

A typical output from the "Design Mode" of "GRAINS" is shown in Table 6.

Once the "GRAINS Design Mode" has been run, "GRAINS Burnback" is run to obtain A_b versus web.

The output from "Burnback" for Case II is shown in Table 7.

"ROCKET" can then be run to obtain the pressure-time and thrust-time profiles. Input data for Case II are shown in Table 8.

A small section of the output from "ROCKET" is shown in Table 9 for Case II.

A summary of the results for the three cases is presented in Table 10 and pressure-time traces are shown in Figure 7.

None of the cases attempted met the desired design values with sufficient accuracy. Further iterations may result in a better solution. However, the star design has considerable sliver with the resulting long tailoff. This section was presented in order to demonstrate the process required to obtain a final grain design.

TABLE 6

OUTPUT FROM "DESIGN MODE" OF "GRAINS"

```
Design Inputs
-------------
Maximum Number of Spokes =  8      Grain Outer Radius =    8.00000
Desired Port Area =    43.89293    Desired Perimeter Factor =    2.48915
RI1 =      3.50000     DEL2 =    -0.25000     FIN2 =     2.00000
SRI1 =     0.25000     DEL1 =     0.10000     FIN1 =     0.25000
SRI2 =     0.25000     DEL4 =     0.10000     FIN4 =     0 25000
FINWEB =   4.00000     WEBSTP =   0.25000     KSPOKE =   0.50000
CIRTOL =   0.10000     PTTOL =    0.10000     PERTOL =   0.10000

Tolerance Limits
----------------
Port Area: Minimum =    39.50364     Maximum =    48.28222
Perimeter: Minimum =    52.61331     Maximum =    64.30515
Circle    : Minimum =  324.00000     Maximum =   396.00000

Web Thickness =   2.5000


N APORT  PERIM   PFAC    RR1    R2     R3     R4     RR5    SR1    SR2    SPKT   PHI1   PHI2   PHI3   PHI4

ERROR  ->  G1 greater than PTTOL, therefore equal
8 51.532 53.535  2.100  3.500  3.750  5.250  5.500  8.000  0.250  0.250  3.107  3.323 26.814  2.729  5.082
8 4[illegible] 5[illegible]  2.314  3.25[illegible]  3.5[illegible]  5.25[illegible]  5.50[illegible]  [illegible]  [illegible]  [illegible]  [illegible]  4.0[illegible] 26.814  2.7[illegible]  4.[illegible]  III
8 42.[illegible] 53.737  2.553  3.000  3.250  5.25[illegible]  5.5[illegible]  8.000  0.250  0.25[illegible]  [illegible]  4.41[illegible] 26.814  2.7[illegible]  [illegible]  II
ERROR  ->  G1 greater than PTTOL, therefore equal
8 37.377 51.222  2.825  2.750  3.000  5.250  5.500  8.000  0.250  0.250  3.274  4.780 26.814  2.729  3.166
ERROR  ->  G1 greater than PTTOL, therefore equal
8 32.521 53.600  3.143  2.500  2.750  5.250  5.500  8.000  0.250  0.250  3.349  5.216 26.814  2.729  2.295
ERROR  ->  G1 greater than PTTOL, therefore equal
ERROR  ->  G2 greater than PERTOL, therefore equal
8 27.651 65.844  3.532  2.250  2.500  5.250  5.500  8.000  0.250  0.250  3.440  5.735 26.814  2.729  1.248
ERROR  ->  Phi4:I  0.0[illegible] less than 0.0
   Spokes: 8  Phi1:  0.1  Phi2:  0.5  Phi3:  0.0
```

TABLE 6 (CONTINUED)

OUTPUT FROM "DESIGN MODE" OF "GRAINS"

```
web Thickness =   2.7500


N APORT  PERIM    PFAC    RR1    R2     R3     R4     RR5    SR1    SR2    SPKT   PHI1  PHI2   PHI3   PHI4

ERROR  ->  S2 greater than PERTOL, therefore equal
8 45.465 48.191  2.016  3.500  3.750  5.000  5.250  8.000  0.250  0.250  3.287  3.320 30.789  2.868  0.53-
ERROR  ->  S2 greater than PERTOL, therefore equal
8 40.285 50.705  1.254  3.250  3.500  5.000  5.250  8.000  0.250  0.250  3.322  4.098 30.789  2.868  0.167
ERROR  ->  Phi4:1  0.01 less than 0.0
   Spokes: 8  Phi1:  0.1  Phi2:  0.5  Phi3:  0.1
ERROR  ->  Phi4:1  0.01 less than 0.0
   Spokes: 8  Phi1:  0.1  Phi2:  0.5  Phi3:  0.1
ERROR  ->  Phi4:1  0.01 less than 0.0
   Spokes: 8  Phi1:  0.1  Phi2:  0.5  Phi3:  0.1
ERROR  ->  Phi4:1 -0.11 less than 0.0
   Spokes: 8  Phi1:  0.1  Phi2:  0.5  Phi3:  0.1
ERROR  ->  Phi4:1 -0.11 less than 0.0
   Spokes: 8  Phi1:  0.1  Phi2:  0.5  Phi3:  0.1
```

TABLE 7

GRAINS "BURNBACK" OUTPUT FOR CASE II

Design Inputs

Maximum Number of Spokes = 0 Grain Outer Radius = 8.00000

Desired Port Area = 42.12200 Desired Perimeter Factor = 2.55300

RI1 =	3.00000	DEL2 =	0.00000	FIN2 =	0.00000
SRI1 =	0.25000	DEL1 =	0.00000	FIN1 =	0.25000
SRI2 =	0.25000	DEL4 =	0.00000	FIN4 =	0.25000
FINWEB =	0.00000	WEBSTP =	0.00000	KSPOKE =	0.00000
CIRTOL =	0.00000	PTTOL =	0.00000	PERTOL =	0.00000

Burnback Inputs

Grain Length = 40.00000 PHI2 = 25.81400 PHI4 = 3.90300

STEP1 = 5. STEP2 = 5. STEP3 = 5. STEP4 = 5. STEP5 = 5.

Number of Spokes = 8

Web Thickness = 2.50000

Web Burned (IN)	Perimeter (IN)	Perimeter Factor	Burn Area (SQ IN)	Port Area (SQ IN)
Burnback of SR1 (Spoke Corner Radius)				
0.000	58.737	2.553	2349.472	42.121
0.050	59.051	2.481	2362.038	45.066
0.100	59.365	2.416	2374.604	48.027
0.150	59.679	2.357	2387.170	51.003
0.200	59.993	2.303	2399.737	53.994
0.250	60.308	2.253	2412.303	57.002
0.250	60.308	2.253	2412.303	57.002
0.465	61.389	2.054	2455.576	71.054
0.679	62.150	1.904	2485.997	84.747
0.894	62.589	1.784	2503.564	97.378
1.108	62.737	1.682	2508.279	110.643
1.323	62.504	1.592	2500.141	122.639

TABLE 7 (CONTINUED)

GRAINS "BURNBACK" OUTPUT FOR CASE II

Burnback of LSPOKE (Spoke Side)

1.323	62.504	1.579	2500.141	124.896
1.395	60.810	1.511	2432.411	128.918
1.468	59.117	1.447	2364.680	132.817
1.540	57.424	1.386	2296.950	136.593
1.613	55.730	1.328	2229.220	140.247
1.686	54.037	1.271	2161.489	143.777

Burnback of remainder to WEB = 0

1.686	54.037	1.271	2161.489	143.777
1.848	52.036	1.189	2081.455	152.320
2.011	51.533	1.146	2061.309	160.305
2.174	51.586	1.119	2063.440	169.198
2.337	51.930	1.099	2077.196	177.625
2.500	52.449	1.085	2097.974	186.124

Zero web Thickness .

Percent of fuel remaining = 9.40

Burnback of remaining fuel

2.832	20.320	0.409	812.798	196.073
2.943	15.430	0.309	617.194	195.651
3.066	10.419	0.208	416.753	199.637
3.200	5.278	0.105	211.107	200.683
3.343	0.000	0.000	0.000	211.059

TABLE 8

INPUT DATA FOR "ROCKET", CASE II

Motor Performance Program IBM-PC version 1.0

INPUT DATA...

MOTOR HAS 1 GRAINS

DESCRIPTION OF GRAIN 1

BURN RATE	0.3846
BURN RATE EXP.	0.4000
PI SUB K	0.2000
BURN REF. PRESS.	1030.2
C STAR	4766.0
C STAR EXP.	0.0061
DENSITY	0.0636
IGNITION TIME	0.0000
DELTA IGN. TIME	0.0000

Normal Input is finished

General Configuration PARAMETERS are ...

Throat AREA	8.8510
Exit AREA	117.2903
Expansion RATIO	13.2517
Total Motor VOLUME	8042.5
AMBIENT Temperature	70.0
AMBIENT Pressure	10.1
Closure BLOWOUT	35.0
Pzero	15.0
Lambda	1.0000
Cd	0.9246
Gamma	1.1952
Throat DESIGN Pressur	1030.2
THROAT EROS. DELAY TI	2000.000

PRESSURE VS. THROAT EROSION RATE

600.	0.000000
1400.	0.000000

WEB	BURN AREA
0.000	2349.472
0.250	2411.952
1.108	2507.744
1.540	2296.212
2.011	2060.360
2.500	2097.107
2.832	812.473
3.342	0.000

TOTAL PROPELLANT WEIGHT= 411.996 IGNITER WEIGHT= 0.00

TABLE 9

SAMPLE OUTPUT FROM "ROCKET", CASE II

TIME	PC	PDOT	FREE VOL	P WEIGHT	WDOTP	ING WT	WDOT	AT	AVAL
7.205	586.14	-655.13	7533.15	379.60	33.81	0.00	35.12	8.851	0.0000
7.225	572.85	-645.45	7543.90	380.28	33.04	0.00	34.33	8.851	0.0000
7.246	559.73	-634.70	7554.42	380.95	32.27	0.00	33.55	8.851	0.0000
7.266	546.85	-624.11	7564.68	381.60	31.52	0.00	32.78	8.851	0.0000
7.287	534.18	-613.15	7574.71	382.24	30.78	0.00	32.03	8.851	0.0000
7.307	521.73	-602.31	7584.50	382.86	30.06	0.00	31.28	8.851	0.0000
7.327	509.51	-591.39	7594.07	383.47	29.35	0.00	30.56	8.851	0.0000
7.348	497.51	-580.60	7603.40	384.07	28.65	0.00	29.84	8.851	0.0000
7.368	485.73	-569.85	7612.51	384.65	27.97	0.00	29.14	8.851	0.0000
7.389	474.17	-559.22	7621.41	385.21	27.30	0.00	28.45	8.851	0.0000
7.409	462.82	-548.69	7630.09	385.75	26.64	0.00	27.77	8.851	0.0000
7.430	451.69	-538.30	7638.56	386.30	25.99	0.00	27.11	8.851	0.0000
7.450	440.77	-528.02	7646.83	386.83	25.36	0.00	26.46	8.851	0.0000
7.471	430.06	-517.90	7654.89	387.34	24.74	0.00	25.82	8.851	0.0000
7.491	419.56	-507.88	7662.76	387.84	24.13	0.00	25.19	8.851	0.0000
7.512	409.26	-498.04	7670.43	388.33	23.53	0.00	24.58	8.851	0.0000
7.532	399.16	-488.28	7677.91	388.81	22.95	0.00	23.97	8.851	0.0000
7.573	379.55	-469.21	7692.32	389.72	21.81	0.00	22.80	8.851	0.0000
7.614	360.71	-450.70	7706.01	390.59	20.72	0.00	21.68	8.851	0.0000
7.655	342.62	-432.72	7719.01	391.42	19.68	0.00	20.60	8.851	0.0000
7.696	325.25	-415.26	7731.35	392.21	18.67	0.00	19.56	8.851	0.0000
7.737	308.59	-398.32	7743.06	392.95	17.71	0.00	18.56	8.851	0.0000
7.778	292.62	-381.89	7754.16	393.66	16.78	0.00	17.61	8.851	0.0000
7.819	277.30	-365.96	7764.68	394.33	15.90	0.00	16.69	8.851	0.0000
7.860	262.63	-350.52	7774.64	394.96	15.05	0.00	15.81	8.851	0.0000
7.901	248.58	-335.56	7784.07	395.56	14.24	0.00	14.97	8.851	0.0000
7.942	235.13	-321.08	7792.99	396.13	13.47	0.00	14.17	8.851	0.0000
7.983	222.27	-307.06	7801.42	396.66	12.72	0.00	13.40	8.851	0.0000
8.024	209.97	-293.50	7809.38	397.17	12.01	0.00	12.66	8.851	0.0000
8.065	198.22	-280.38	7816.90	397.65	11.34	0.00	11.96	8.851	0.0000
8.106	186.99	-267.70	7823.99	398.10	10.69	0.00	11.28	8.851	0.0000
8.147	176.28	-255.44	7830.67	398.52	10.07	0.00	10.64	8.851	0.0000
8.188	166.21	-221.51	7836.97	398.92	9.54	0.00	10.04	8.851	0.0000
8.249	155.54	-133.82	7845.96	399.50	9.09	0.00	9.40	8.851	0.0000
8.341	145.01	-98.55	[illegible]	400.31	8.55	0.00	8.76	8.851	0.0000
8.464	133.96	-84.16	7874.60	401.32	7.91	0.00	8.10	8.851	0.0000
8.628	120.86	-76.13	7893.98	402.55	7.14	0.00	7.31	8.851	0.0000
8.792	108.87	-69.82	7911.45	403.66	6.43	0.00	6.59	8.851	0.0000
8.956	97.89	-64.30	7927.17	404.66	5.78	0.00	5.93	8.851	0.0000
9.119	87.79	-58.96	7941.29	405.56	5.19	0.00	5.32	8.851	0.0000
9.324	76.36	-52.70	7956.89	406.55	4.51	0.00	4.63	8.851	0.0000
9.529	66.16	-47.05	7970.42	407.41	3.91	0.00	4.02	8.851	0.0000
9.775	55.39	-40.86	7984.27	408.30	3.27	0.00	3.37	8.851	0.0000
10.062	44.65	-34.30	7997.55	409.14	2.64	0.00	2.72	8.851	0.0000
10.389	34.48	-27.91	8009.53	409.90	2.04	0.00	2.10	8.851	0.0000
10.799	24.48	-21.16	8020.66	410.61	1.45	0.00	1.50	8.851	0.0000
11.311	15.62	-11.34	8030.00	[illegible]	0.91	0.00	0.94	8.851	0.0000

TABLE 9 (CONTINUED)

SAMPLE OUTPUT FROM "ROCKET", CASE II

AVERAGE PRESSURE FROM 0.000 SEC. TO 0.000 SEC. WAS 0.0 PSI
MAXIMUM PRESSURE WAS 1017.2 PSI

TIME	PC	PE	F	INT PC	INT F	WEIGHT	CF	WT EXH	ISP
7.573	379.55	3.33	4368.49	6506.01	86251.34	389.72	1.300	388.90	221.78
7.614	360.71	3.30	4111.61	6521.17	86425.01	390.59	1.288	389.81	221.71
7.655	342.62	3.34	3872.89	6535.58	86588.55	391.42	1.277	390.67	221.64
7.696	325.25	3.37	3646.54	6549.25	86742.58	392.21	1.267	391.50	221.57
7.737	308.59	3.41	3429.23	6562.24	86887.49	392.95	1.256	392.28	221.49
7.778	292.62	3.45	3221.64	6574.55	87023.70	393.66	1.244	393.02	221.42
7.819	277.30	3.48	3023.47	6586.22	87151.60	394.33	1.232	393.72	221.35
7.860	262.63	3.52	2834.40	6597.28	87271.57	394.96	1.219	394.38	221.28
7.901	248.58	3.56	2654.13	6607.75	87383.96	395.56	1.206	395.02	221.22
7.942	235.13	3.60	2482.38	6617.65	87489.17	396.13	1.193	395.61	221.15
7.983	222.27	3.64	2328.52	6627.02	87587.70	396.66	1.184	396.18	221.08
8.024	209.97	3.68	2183.17	6635.87	87680.10	397.17	1.175	396.71	221.02
8.065	198.22	3.72	2044.56	6644.23	87766.70	397.65	1.166	397.22	220.95
8.106	186.99	3.77	1913.34	6652.12	87847.76	398.10	1.156	397.69	220.89
8.147	176.28	3.81	1788.37	6659.56	87923.57	398.52	1.145	398.14	220.84
8.188	166.21	3.86	1671.44	6666.59	87994.43	398.92	1.136	398.56	[illegible]
8.249	155.54	3.91	1548.19	6676.48	88093.34	399.50	1.125	399.16	[illegible]
8.341	145.01	3.96	1427.11	6690.31	88230.45	400.31	1.112	400.00	220.58
8.464	133.96	4.03	1300.94	6707.45	88396.05	401.32	1.097	401.03	220.43
8.528	120.86	4.11	1152.50	6728.32	88593.04	402.55	1.077	402.30	220.23
8.792	103.87	4.20	1017.33	6747.14	88776.64	403.66	1.055	403.44	220.05
8.956	97.53	4.29	895.70	6764.08	88938.61	404.66	1.034	404.48	219.88
9.119	87.73	4.38	784.56	6779.29	89071.26	405.56	1.010	405.38	219.72
9.324	76.36	4.51	660.20	6796.10	89219.20	406.55	0.977	406.40	219.53
9.529	66.16	4.64	550.78	6810.70	89343.21	407.41	0.941	407.29	219.36
9.775	55.39	4.81	437.17	6825.63	89464.61	408.30	0.892	408.20	219.17
10.062	44.65	5.02	325.18	6839.98	89574.05	409.14	0.825	409.07	218.97
10.389	34.48	5.28	224.00	6852.94	89664.15	409.90	0.734	409.86	218.77
10.799	24.48	5.66	126.70	6865.02	89738.01	410.61	0.585	410.60	218.55
11.311	15.62	6.19	43.51	6875.28	89779.58	411.20	0.315	[illegible]	218.33

TABLE 10

SUMMARY OF RESULTS FOR THREE GRAIN DESIGNS
D = 16 IN, L = 40 IN

GRAIN 3	I	II	III
No. of pts.	6	8	8
web, in	2.5	2.5	2.5
Required $\dot{r}$ (in/sec)	0.385	0.385	0.385
P.F.	2.218	2.553	2.314
Port Area, in^2	42.48	42.12	46.88
RR1, in	2.750	3.0	3.25
ϕ_2	26.81	26.81	26.81
ϕ_4	10.03	3.90	4.54
SR1, in	0.5	0.25	0.25
SR2, in	0.25	0.25	0.25
P_{max}, psia	990.	1,017.	952.
$F_{max} lb_f$	12,464.	13,702.	12,749.
$\bar{F}$ lb_f	6,956.	7,945.	7,738.
I_T, lb_f-sec	91,151.	89,780.	87,690.
$t_{burn\ total}$, sec	13.1	11.3	11.2

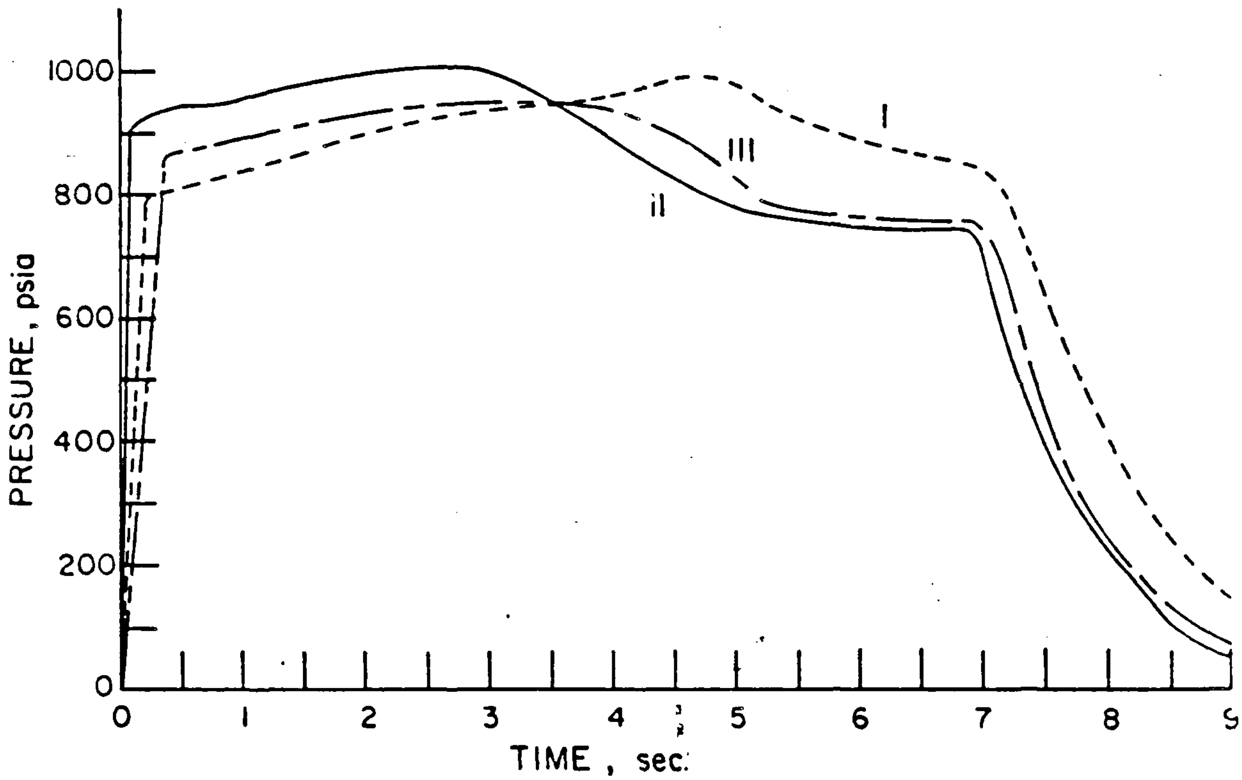


FIGURE 7.

PRESSURE-TIME PROFILES FROM "ROCKET"

X. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

An interactive computer code for the preliminary design of solid rocket propellant rocket motors ("SPRMD") was successfully developed and its use was demonstrated through a design example. The code incorporated several existing PC codes and AGARD recommended empirical expressions for losses in aluminized propellant systems.

B. RECOMMENDATIONS

Further modifications to improve the capabilities of the code are essential in order to make "SPRMD" a more effective preliminary design tool.

- There is only one grain configuration (radially spoked star grains) incorporated into "SPRMD."
  It is necessary to add other options for grain design in order to increase the grain design flexibility. One possibility would be to incorporate the Grain Design and Internal Ballistics (GDB) module of Solid Performance Prediction Code (SPP) (Ref. 3).
- "SPRMD" also does not currently incorporate any plotting capabilities. There is definitely a need for this program to have plotting capabilities for the grain design results and the thrust and pressure versus time profiles.

- "FLYIT" was not incorporated into the program, although "SPRMD" was structured to have flight simulation capabilities. It is also recommended that "FLYIT" be incorporated into "SPRMD" as another enhancement.
- "SPRMD" is a primative type of preliminary solid propellant rocket motor design code. Further enhancements, such as propellant searching, screen editing, and increased solution speed for "MICROPEP" and "ROCKET" would make this code more user friendly.

APPENDIX

```
      program SPRMD
c***************************************************************
c                                                              *
c SOLID PROPELLANT ROCKET MOTOR DESIGN                         *
c THIS PROGRAM (SPRMD) SOLVE THE PRELIMINARY DESIGN PROBLEM*
c OF SOLID PROPELLANT ROCKET MOTOR. SUBROUTINES  INCLUDES: *
c (1) INPUT0: BASIC INPUT FOR DESIGN                           *
c (2) INPUT1: CHEMICAL COMPOSITION INPUT                       *
c (3) MICROPEP: PERFORME THE PROPELLANT EVALUTION              *
c (4) PFE: SIZE THE THROAT OF NOZZLE PORT AREA OF GRAIN ETC*
c (5) LOSCF: CALCULATE THE THRUST COEFFICIENT                  *
c (6) INPUT2: STAR GRAIN INPUT                                 *
c (7) GRAINS: PERFORM THE SPOKED STAR GRAINS DESIGN            *
c (8) RCKTIN: INPUT THE DATA FOR THRUST AND PRESSURE PRO-      *
c             FILE CALCULATION                                 *
c (9) ROCKET: COMPUTED THE THRUST AND PRESSURE PROFILE         *
c***************************************************************
c
      real favg,tb,l,d,meop,pc,rrqd,pamb,tamb,h,ld,webrqd,wf,m
      common /micrp/ a(12,12),kr(20),amat(10,12),jat(12),in,is,
     1fie(10,6),ie(10,6),alp(12),w27,n,dh(10),rho(10),
     2wate(10),w1(6),w43,ig,np,vnt(201),w47,name,ser,
     3floor,itag(100),wing(10)
c     integer
      character*10 yes
c***************************************************************
c                                                              *
c BASIC INPUT FOR MOTOR DESIGN                                 *
c                                                              *
c***************************************************************
1000  call input0(favg,tb,l,d,meop,pc,rrqd,pamb,tamb,h,fmin,tpmax,tpnom,
     *tpmin,pik)
      write(*,1)favg
    1 format(1x,'AVERAGE THRUST, LBF................... ',f12.1)
      write(*,2)tb
    2 format(1x,'BURN TIME, SEC......................... ',f12.2)
      write(*,12)fmin
   12 format(1x,'MINIMUM THRUST REQUIRED, LBF.......... ',f12.6)
      write(*,3)l
    3 format(1x,'MOTOR LENGTH, INCH.................... ',f12.2)
      write(*,4)d
    4 format(1x,'MOTOR DIAMETER, INCH.................. ',f12.2)
      write(*,5)meop
    5 format(1x,'MAXIMUM EXPECTED OPERATING PRESSURE... ',f12.1)
      write(*,13)tpmax
   13 format(1x,'TPMAX, F..............................',f12.1)
      write(*,131)tpnom
  131 format(1x,'TPNOM, F.............................. ',f12.1)
      write(*,132)tpmin
  132 format(1x,'TPMIN, F.............................. ',f12.1)
      write(*,14)pik
   14 format(1x,'TEMPERATURE SENSITIVITY, %/F.......... ',f12.3)
      write(*,6)pc
    6 format(1x,'NOMINAL CHAMBER PRESSURE, PSI........ ',f12.1)
      write(*,7)rrqd
    7 format(1x,'REQUIRED BURNING RATE, IN/SEC......... ',f12.3)
      write(*,8)h
    8 format(1x,'ALTITUDE, FT.......................... ',f12.0)
      write(*,9)pamb
    9 format(1x,'AMBIENT PRESSURE, PSI.................',f12.2)
```

```
      write(*,11)tamb
 11   format(1x,'AMBIENT TEMPERATURE, R................',f12.1)
      write(*,10)
10    format(1x,'ANY CORRECTION OF BASIC INPUT? Y-YES, N-NO... ')
      read(*,20)yes
20    format(a10)
      if(yes.eq.'y'.or.yes.eq.'Y') go to 1000
c
c*************************************************************
c                                                           *
c INPUT THE DATA NEEDED BY MICROPEP                         *
c                                                           *
c*************************************************************
c
c
1010  call input1(pc,pamb)
      write(*,40)
40    format(1x,'Are there any corrections of input for micropep?,y-yes
     *n-no ')
      read(*,20)yes
      if(yes.eq.'Y'.or.yes.eq.'y') go to 1010
c
c*************************************************************
c                                                           *
c COMPUTE THERMAL EQUILIBRIUM, PROPELLANT CHARACTERISTIC,   *
c AND ETC.(USING MICROPEP, UNTILL GOOD RESULTS ARE REACHED)*
C                                                           *
C*************************************************************
c
      call micropep
      write(*,50)
50    format(1x,'Are the results good? y-yes n-no  ')
      read(*,20)yes
      if(yes.eq.'y'.or.yes.eq.'Y') then
         go to 1020
      else
55       write(*,60)
60       format(1x,'(1)  Change the Basic Inputs of motor.')
         write(*,70)
70       format(1x,'(2)  Change the Input data for Micropep.')
         write(*,80)
80       format(1x,'     type 1, or 2, if 1 is chosen it is necessary')
         write(*,90)
90       format(1x,'     that Micropep shall be run once again!!!!')
         read(*,100)i
100      format(i1)
         if(i.ne.1.and.i.ne.2)  then
            write(*,*)'      Error input!! 1, or 2 must be typed'
            go to 55
         else
            if(i.eq.1) go to  1000
            if(i.eq.2) go to  1010
         endif
      endif
c*************************************************************
c                                                           *
c CALCULATE THE LOSS PERFORMANCE & COMPUTE THE DATA  FOR    *
c GRAIN DESIGN                                              *
c                                                           *
c*************************************************************
```

APPENDIX

```
c
1020  call pfe(favg,tb,l,d,pc,cfc,ath,etacf,m,vp,vl,dpe,aport,
     *pfac,isp,cfth,webrqd,wf,tc,rrqd,ae,etac,xx)
      write(*,50)
      read(*,20)yes
      if(yes.eq.'y'.or.yes.eq.'Y') then
         go to 1030
      else
   75    write(*,60)
         write(*,70)
         write(*,71)
   71    format(1x,'(3)  Change nozzle data.')
         write(*,72)
   72    format(1x,'     type 1, 2, or 3   ')
         read(*,100)i
         if(i.lt.1.and.i.gt.3)  then
            write(*,*)'Error input!! 1, 2 or 3 must be typed'
            go to 75
         else
            if(i.eq.1) go to  1000
            if(i.eq.2) go to  1010
            if(i.eq.3) go to  1020
         endif
      endif
c***************************************************************
c                                                             *
c CHOOSE THE APPROPRIATE GRAIN DESIGN                         *
c NOTE: OPTION 2, 3, 4, 5, 6, 7 ARE NOT AVAILABLE. INPUT      *
c YOUR OWN DESIGN BY SUBSTITUTE THE WRITE AND GOTO STATE-     *
c MENTS IN THE CORRESPONDING BLOCK                            *
c                                                             *
c***************************************************************
1030  write(*,110)
110   format(1x,'Choose one of the following grain configurations')
      write(*,120)
120   format(1x,'(1) Star grain')
      write(*,130)
130   format(1x,'(2) Optional grain ')
      write(*,140)
140   format(1x,'(3) Optional grain ')
      write(*,150)
150   format(1x,'(4) Optional grain ')
      write(*,160)
160   format(1x,'(5) Optional grain ')
      write(*,170)
170   format(1x,'(6) Optional grain ')
      write(*,180)
180   format(1x,'(7) Optional grain ')
      write(*,190)
190   format(1x,'Type 1, 2, 3, .....etc.')
      read(*,*)i
      if(i.lt.1.or.i.gt.7) then
        write(*,200)
200     format(1x,'Error input!! must type 1, 2, 3,.....,7')
        go to 1030
      else
       if(i.eq.1) then
         rr5=d/2.0
         write(*,*)'Run "DESIGN MODE"  '
 1050    call input2(pfac,aport,webrqd,rr5,l,tb,xx)
```

```
          call grains
          write(*,*)'Are the results are acceptable? y-acceptable n-not
     *acceptable type y or n..'
          read(*,20)yes
          if(yes.eq.'n'.or.yes.eq.'N') then
            go to 1050
          else
            go to 1055
          endif
 1055     rr5=d/2.0
          write(*,*)'Run the "BURNBACK MODE" to get web vs. burning area'
          call input2(pfac,aport,webrqd,rr5,1,tb,xx)
          call grains
          write(*,*)'Are the results are acceptable? y-acceptable n-not
     *acceptable type y or n..'
          read(*,20)yes
          if(yes.eq.'n'.or.yes.eq.'N') then
            write(*,*)'Do you want to try another star grain? y/n   '
            read(*,20)yes
            if(yes.eq.'n'.or.yes.eq.'N') then
              go to 1030
            else
              go to 1050
            endif
          else
            go to 1060
          endif
        else
          if(i.eq.2) then
             write(*,280)
 280         format(1x,'Input your own grain design package')
             go to 1030
          else
            if(i.eq.3) then
               write(*,290)
 290           format(1x,'Input your own grain design package')
               go to 1030
            else
              if(i.eq.4) then
                write(*,300)
 300            format(1x,'Input your own grain design package')
                goto 1030
              else
                if(i.eq.5) then
                   write(*,310)
 310               format(1x,'Input your own grain design package')
                   go to 1030
                else
                  if(i.eq.6) then
                    write(*,320)
 320                format(1x,'Input your own grain design package')
                    go to 1030
                  else
                    if(i.eq.7) then
                      write(*,330)
 330                  format(1x,'Input your own grain design package')
                     go to 1030
                    endif
                  endif
                endif
```

```
          endif
        endif
      endif
    endif
   endif
c
c***********************************************************
c                                                          *
c INPUT THE DATA FOR THRUST & PRESSURE  VERSUS TIME PROFILE*
c CALCULATION                                              *
c                                                          *
c***********************************************************
 1060 write(*,*)'The following processes is to calculate the "THRUST VS.
     * TIME" and "PRESSURE VS. TIME" profiles'                '
      va=(d/2.0)**2.0*3.14159*l
      rrqd=webrqd/tb
      write(*,106)rrqd
  106 format(lx,'Required burning rate for new web, IN/SEC.. ',f6.4)
 1070 call rcktin(ath,ae,va,tamb,pamb,pc,etacf,rrqd,pik,etac)
      write(*,*)'Any change for the input? y-yes, n-no     '
      read(*,20)yes
      if(yes.eq.'y'.or.yes.eq.'Y') then
        go to 1070
      else
        go to 1080
      endif
c
c***********************************************************
c                                                          *
c COMPUTE THE PRESSURE VS. TIME  &  THRUST VS.TIME PROFILE *
c OF ROCKET MOTOR                                            *
C                                                          *
c***********************************************************
c
 1080 call rocket
c
      write(*,340)
340   format(lx,'Are the results acceptable? y-yse, n-no......')
      read(*,20)yes
      if(yes.eq.'y'.or.yes.eq.'Y')then
         go to 9999
      endif
      write(*,350)
350   format(lx,'Do you want to proceed? y-yes, n-no..........')
      read(*,20)yes
      if(yes.eq.'N'.or.yes.eq.'n') then
         write(*,*)'Terminated on the request of user '
         go to 9999
      endif
1040  write(*,360)
360   format(lx,'Suggest to change the following data')
      write(*,380)
380   format(lx,'(1) Change the Grain Design.')
      write(*,390)
390   format(lx,'(2) Change the Ingredients of the propellant.')
      write(*,400)
400   format(lx,'(3) Change the Basic Design Requirements.')
      write(*,*)'(4) Re-run the "ROCKET"'
      write(*,*)' Type 1, 2 or 3 for selection'
      read(*,100)i
```

```
      if(i.ne.1.and.i.ne.2.and.i.ne.3) then
         write(*,*)' Error input you have to type 1, 2 or 3'
         go to 1040
      else
         if(i.eq.1) then
           go to 1030
         else
           if(i.eq.2) then
             go to 1010
           else
           if(i.eq.3) then
             go to 1000
           else
             if(i.eq.4) go to 1070
           endif
           endif
         endif
      endif
c************************************************************
c                                                           *
c INPUT THE TRAJECTORY DATA  (FUTURE WORK)                  *
c                                                           *
c************************************************************
c
c      call flyin
c
c************************************************************
c                                                           *
c CALCULATE THE TRAJECTORY OF ROCKET MOTOR (FUTURE WORK)    *
c                                                           *
c************************************************************
c
c     call flyit
9999  end
```

APPENDIX

```
      subroutine Input0(favg,tb,l,d,meop,pc,r,pamb,tamb,h,fmin,tpmax,
     *tpnom,tpmin,pik)
c***************************************************************
c                                                              *
c THIS SUBROUTINE IS FUNCTION AS A BASIC INPUT FOR SOLID       *
c PROPELLANT ROCKET MOTOR DESIGN                               *
c                                                              *
c***************************************************************
c
c variable definition:
c     ********* (ballistic performance) *********
c     favg    : average thrust (F12.6)
c     tb      : burning time of motor (F12.6)
c     fmin    : minimum thrust (F12.6)
c     ********* (mission/vehicle) ***************
c     l       : motor length (F12.6)
c     d       : motor diameter (F12.6)
c     meop    : maximum expeted operating pressure (F12.6)
c     va      : motor volume avaiable for propellant (F12.6)
c     tpmax, tpnom, tpmin
c     pmaxtp  : meop * (1-reproducibility tolerance) (F12.6)
c     pc      : norminal chamber pressure (F12.6)
c     rptol   : reproducibility tolerance (F12.6)
c     r       : burning rate (F12.6)
c     web     : web of grain (F12.6)
c     pik     : temperature sensitivity of propellant(F12.6)
c     h       : altitude(F12.6)
      real favg,tb,fmin,l,d,meop,va,dltp,pmaxtp,pc,rptol,r,web,h
c     open a file which can hold the basic input data
      open(2,file='input0.dat',access='sequential',status='unknown')
c     input basic data
      write(*,*)'All input data in decimal format'
      write(*,*)'Specify the average thrust, LBF........ '
      read (*,10)favg
10    format(f12.6)
      write(2,10)favg
      write(*,*)'Input burning time of motor, SEC....... '
      read(*,10)tb
      write(2,10)tb
      write(*,*)'Input minimum thrust required, LB...... '
      read(*,10)fmin
      write(*,*)'Input motor length, INCHES............. '
      read(*,10)l
      write(2,10)l
      write(*,*)'Input motor diameter, INCHES........... '
      read(*,10)d
      write(2,10)d
      write(*,*)'Input maximum expected '
      write(*,*)'operating pressure, PSI................ '
      read(*,10)meop
      write(2,10)meop
      write(*,*)'Input Tpmax, F......................... '
      read(*,10)tpmax
      write(2,10)tpmax
      write(*,*)'Input Tpnom, F......................... '
      read(*,10)tpnom
      write(2,10)tpnom
      write(*,*)'Input Tpmin, F......................... '
      read(*,10)tpmin
      write(2,10)tpmin
```

APPENDIX

```
      write(2,10)dltp
      write(*,*)'Input reproducibility tolerance, %..... '
      read(*,10)rptol
      write(2,10)rptol
      rptol=rptol/100.0
      pmaxtp=meop*(1.0-rptol)
      rptol=rptol*100.0
      r=(d/4)/tb
      write(*,20)r
      write(2,10)r
20    format(1x,'Estimated repuired burn rate............ ',f12.3,
     *' INCH/SEC')
      write(*,30)
c     The following calculation of ambient temperature and pressure at
c     certain altitude is directly copied from NWC Technical Memorandum
c     4757 subroutine atmstd based on U.S Standard Atmosphere, 1976.
30    format(1x,'Input the required design altitude of rocket motor in F
     *T                    ......................................
     *')
      read(*,10)h
      write(2,10)h
      if(h-36152.)1,1,2
1     tamb=518.67-0.003559969*h
      ro=0.00237696*exp(-(0.028800339*(h*.001)+0.000132317*(h*.001)**2))
      go to 7
2     if(h-65825.)3,3,4
3     tamb=389.97
      ro=0.00237696*0.29697*exp(-0.04771923*(h-36152.)*.001)
      go to 7
4     if(h-105511.)5,5,6
5     tamb=354.142+0.0005442926*h
      ro=0.00237696*0.0718594*exp(-(0.049130937*(h-64825.)*.001
     &-0.000034655*((h-65825.)*.001)**2))
      go to 7
6     tamb=251.4114+0.00151794*h
      ro=0.00237696*0.0107993*exp(-(h-105511.)/21320.)
7     pamb=11.9203*ro*tamb
      write(2,10)pamb
      write(2,10)tamb
c     end of calculation of tempressure and temperature at design alt.
c
      write(*,*)'Based on available requirements, make an initial'
      write(*,*)'selection of propellant. Then input the temperature'
      write(*,*)'sensitivity of the propellant, %/F........... '
      read(*,40)pik
      write(2,40)pik
40    format(f12.6)
      pik=pik/100.0
      pc=pmaxtp*exp(pik*(tpnom-tpmax))
      pik=pik*100.0
      write(2,10)pc
      close(2,status='keep')
      return
      end
```

APPENDIX

```
      subroutine input1(pc,pamb)
c*************************************************************
c                                                            *
c THE PRIMARY PURPOSE OF THIS SUBROUTINE IS TO ESTABLISH     *
c The DATA FILE WHICH CAN BE ACCESSED BY MICROPEP SUB-       *
c TINE                                                       *
c                                                            *
c*************************************************************
c
      common /micrp/ a(12,12),kr(20),amat(10,12),jat(12),in,is,fie(10,6)
     1,ie(10,6),alp(12),w27,n,dh(10),rho(10),wate(10),w1(6),w43,ig,np,
     2vnt(201),w47,name,ser,floor,itag(100),wing(10)
      common/chara/block,je,aspec,specie
      character*8 specie(200)
      character*30 block(10),ofile
      character*2 je(10,6),aspec(12)
      common/moon/tstest,te,irun,iounit,iend
      character*30 blok(10)
      character*10 user
      dimension jie(10,6)
      integer nic,niu,irun,alt,kr
      real denexp
c
c     open the file as formatted sequential as unit 3
      open(3,file='input.dat',access='sequential',status='unknown')
c  input the case name from the keyboard
      write(*,*)' Start to input the selected propellant data     '
      write(*,*)' '
      write(*,1)
    1 format(1x,'Specify the case name (format A10)......   ')
      read(*,'(a)')user
      write(3,10)user
10    format(1a10)
      irun=2
c  input the number of ingredients, supplied ingredients, and runs
      write(*,*)'Input the number of ingredients? (format I5)....'
      read(*,*)nic
      write(*,*)'Input the number of user suppplied ingredients (I5)...'
      read(*,*)niu
      write(3,25)nic,niu,irun
20    format(2i5)
25    format(3i5)
c  Input density exponent which is used in conjunction with control
c  option 3.
      write (*,*)'Input the density exponent.'
      write(*,*)'For tactical missiles or first stages use 1.0'
      write(*,*)'Use 0.7 for second stages and 0.2 for third stages'
      write(*,*)'(format F10.0).................... '
      read(*,40)denexp
      write(3,40)denexp
40    format(f10.5)
c
c  input control option store it in kr(20)
c     print out the options on the screen
      write(*,'(a)')'There are five options listed below for your'
      write(*,*)'selection. Type 1, 2, 3, 4, 5---'
      write(*,*)'Normal input for design is 3'
      write(*,'(a)')'   1. For chamber and exit calculations.'
      write(*,'(a)')'   2. For chamber only calulations.'
      write(*,'(a)')'   3. For boost performance and nozzle design.'
```

```
      write(*,'(a)')'   4. For equilibrium calculations at specified T'
      write(*,'(a)')'      and P. Use T in Kelvin at Pe location and P'
      write(*,'(a)')'      in psia at the normal Pc location.'
      write(*,'(a)')'   5. Other options ( Check README for special'
      write(*,'(a)')'      alternatives).'
      write(*,*)'Use format I1 '
      read(*,45)alt
45    format(i1)
      if (alt.lt.5) then
              kr(2) = 0
              kr(4) = 0
              kr(5) = 0
              kr(6) = 0
              kr(8) = 0
              if (alt.eq.1) then
                      kr(1) = 0
                      kr(3) = 0
                      kr(7) = 0
                 else
                 if (alt.eq.2) then
                         kr(1) = 1
                         kr(3) = 0
                         kr(7) = 0
                    else
                    if (alt.eq.3) then
                            kr(1) = 0
                            kr(3) = 1
                            kr(7) = 0
                       else
                            kr(1) = 0
                            kr(3) = 0
                            kr(7) = 1
                    endif
                 endif
              endif
              write(3,50)(kr(i),i = 1,8)
50            format(8i1)
         else
         write(*,'(a)')' Input your requirements for calculation--'
         read(*,60)(kr(i),i = 1,15)
         write(3,60)(kr(i),i = 1,15)
60       format (15i1)
      endif
c     input the serial number of ingredients found in JANNAF.DAT table
c
      write(*,70) nic
70    format(1x,'Input ',i2,' ingredients needed for calculation')
      do 100 i = 1,nic
         write(*,80)i
80       format(1x,'Serial # of number ',i5,' ingredient(I5)...    ')
         read(*,90)itag(i)
90       format(i5)
100   continue
      write (3,120)(itag(i), i = 1,nic)
120   format(10i5)
c  Input user supplied ingredients
      if (niu.gt.0) then
      do 170 i = 1,niu
         write(*,'(a)')'Specify ingredient name use caps lock (A30)    '
         read(*,130)blok(i)
```

```
         write(*,'(a)')'Specify composition of ingredient max = 6'
         write(*,'(a)')' use (6(I3,A2))'
         write(*,'(a)')' #############################**'
         read(*,140)(jie(i,l),je(i,l),l=1,6)
         write(*,'(a)')' Specify heat of formation, and density (f5.0,f6
     *.0)    '
         read(*,150)dh(i),rho(i)
         write(3,160)blok(i),(jie(i,l),je(i,l),l=1,6),dh(i),rho(i)
130      format(a30)
140      format(6(i3,a2))
150      format(f5.0,f6.0)
160      format(a30,6(i3,a2),f5.0,f6.0)
170   continue
      else
         write (*,'(a)')' No user supplied ingredients  '
      endif
      if (irun.lt.2) then
          irun = 2
      endif
      status=0
      do 220 i = 1,irun
         if(status.eq.0) then
         wl(5)=250
c        write (*,'(a)')' Specify the desired chamber pressure (F12.4) '
c        read (*,180)wl(5)
         wl(6)=pamb
c        write (*,'(a)')' Specify the exhaust pressure (F12.4)'
c        read (*,190)wl(6)
         ni=nic+niu
         do 175 j =1,ni
         write(*,172)itag(j)
172      format(1x,'Input the weight % of ingredient',i5,'(F10.4)...')
         read(*,190)wing(j)
175      continue
180      format(f10.4)
190      format(f10.4)
         write(3,210)wl(5),wl(6),(wing(j),j=1,10)
210      format(12f10.4)
         status=1
         else
         wl(5)=pc
         write(3,210)wl(5),wl(6),(wing(j),j=1,10)
      endif
220   continue
      close(3)
      return
      end
```

APPENDIX

```
      subroutine pfe(favg,tb,l,d,pc,cfc,ath,etacf,m,vp,vl,dpe,aport,
     +pfac,isp,cfth,webrqd,wf,tc,r,ae,etac,xx)
c*************************************************************
c                                                            *
c THE MAIN PURPOSE OF THIS SUBROUTINE IS TO CALCULATE THE    *
c PERFORMANCE LOSS AND THE DATA FOR GRAIN DESIGN             *
c                                                            *
c*************************************************************
c variable definitions:
c     favg:         Average thrust from basic input
c     cfth:         thrust coefficient from micropep
c     cfc :         corrected cf
c     pc  :         chamber pressure from basic input
c     d   :         motor diameter from basic input
c     l   :         motor length from basic input
c     aththth:      theoretical throat area
c     ath  :        throat area
c     atht :        tempory result of throat
c     dthth:        theoretical throat diameter
c     etacf:        thrust coefficient efficiency
c     dif  :        for convergence test of throat size
c     va   :        vol avaiable for propellant
c     tb   :        burn time
c     vp   :        propellant vol.
c     vl   :        volumetric loading factor
c     aport:        port area
c     dpe  :        equivalent circular port diameter
c     pwet :        wetted perimeter
c     j    :        throat-to-port area ratio
c     rhop :        propellant mass density
c     mw   :        molecular weight of propellant
c     mf   :        mole fraction of condensed phase
c     webrqd:       required web
c     wf   :        web fraction
c     pfac :        perimeter factor
c     character*80 fname
c
      real favg,cfth,cfc,pc,d,l,athth,ath,dthth,etacf,dif,tb,vp,vl,va,j,
     +aport,dpe,rhop,a,nn,rstd,rpc,web,webrqd,wf,m,pwet,pfac,isp,ispm,at
     +ht,mw,tc,ispth,ld,mf
      character*1 yes
c The purpose of this common instruction is to transfer w43(propellant
c density) from MICROPEP: SUBROUTINE PUTIN(LE)
      common /micrp/ a(12,12),kr(20),amat(10,12),jat(12),in,is,
     1fie(10,6),ie(10,6),alp(12),w27,n,dh(10),rho(10),
     2wate(10),wl(6),w43,ig,np,vnt(201),w47,name,ser
     3,floor,itag(100),wing(10)
C The purpose of this common instruction is to transfer the CHAMBER PRE-
C SSURE & MOLECULAR WEIGHT from MICROPEP SUBROUTINE OUT.
      common /tcf/ tfc,rmwtc
c Transfer SPECIFIC IMPULSE, CHARACTERISTISC VEL., CODFF. OF THRUST &
C OPTIMUN EXPANSION RATIO from MICROPEP SUBROUTIN DESING.
      common /socf/ spi(2),cf(2),oex(2)
c calculate the theoretical throat area
      pi=3.14159
c transfer cfth from "MICROPEP"
      cfth=cf(2)
c initialize cfc
      cfc=cfth
      athth=favg/(cfth*pc)
```

APPENDIX

```
c initialize ath=athth
      ath=athth
c calculate theoretical throat diameter
      dthth=sqrt(ath/pi)*2
      write(*,888)cfth,athth,dthth
 888  format(1x,'Cfth=',f5.3,'  Athth=',f6.3,'  Dthth=',f5.3)
c calculate the etacf
      write(*,*)'Input the nozzle half angle'
      write(*,*)'alpha, DEGREES (F5.2)................. '
      read(*,*)alpha
      pi=3.14159
      alpha=(alpha/180.0)*pi
      write(*,*)'Input the nozzle exit half angle'
      write(*,*)'thetaex, DEGREES (F5.2)............... '
      read(*,*)thetaex
      thetaex=(thetaex/180.0)*pi
      write(*,*)'Input radial erosion rate of the throat in IN/SEC'
      write(*,*)'assumed to start at time zero (FORMAT 0.XXXX)....'
      read(*,*)erosr
      write(*,*)'Input the mole fraction of condensed'
      write(*,*`'phase in (MOLE/100 g)................. '
      read(*,*)mf
      write(*,*)'Input the material of nozzle:'
      write(*,*)'1-steel nozzle, 0-ordinary nozzle..... '
      read (*,*)nozle
      atht=ath
      do 9 i=1,50
C calculate the loss mechanism
      call loscf(etacf;tb,ath,pc,alpha,thetaex,erosr,mf,nozle)
c correct the trust coefficient
      cfc=cfth*etacf
c calculate the corrected throat area
      ath=favg/(cfc*pc)
      write(*,8888)etacf,cfc,ath
8888  format(1x,'Etacf=',f12.3,'  Cfc=',f12.3,'  Ath=',f12.3)
c calculate the convergence athth and ath
      dif=abs((atht-ath)/atht)*100.0
      if (dif.gt.1.0) then
         atht=ath
      else
         go to 91
      endif
9     continue
91    va=pi*(d/2)**2*l
      etac=0.93
      vol=pi*(d/4)**2*l
c Density and molecular weight com from "MICROPEP"
      rhop=w43
      mw=rr.wtc
      tc=tfc
      ispth=spi(2)
      write(*,98)rhop,mw,tc,ispth
98    format(' Rhop=',f12.3,'  MW=',f12.3,'  Tc=',f12.1,'  Ispth=',f12.2
     *)
99    isp=ispth*etacf*etac
      m=favg/isp
      vp=favg*tb/(isp*rhop)
      vl=vp/va
      write(*,87)VP,VA,VL
87    format(' VP=',F12.1,'  VA=',F12.1,'  VL=',F12.3)
```

```
      dpe=sqrt(1.0-vl)*d
      aport=(pi/4)*d**2*(1-vl)
      tres=(aport*l)*pc*mw/(m*12*1545*(tc+459.6))
      tres=tres*1000.0
      write(*,88)tres
88    format(' RESIDENCE TIME IN MSEC.................... ',f12.3)
      write(*,89)etac
89    format(' COMBUSTION EFFICIENCY (etac)............ ',f12.3)
      write(*,*)'Change etac? y-yes, n-no............... '
      read(*,20)yes
      if(yes.ge.'y'.or.yes.eq.'Y') then
        write(*,*)'Input etac............................ '
        read(*,30)etac
        vol=aport*l
        go to 99
      endif
c calculate the nozzle exit area
      ae=ath*oex(2)
c calculate the required perimeter factor
      write(*,19)pc
   19 format(1x,'Input the C* with Pcnom=',f10.1,' (F10.4)..... ')
      read(*,*)cstar
      xx=(pc*ath*32.17)/(cstar*rhop*l*3.14159*dpe)
   18 write(*,*)'Input the estimated required web (INCHES) (F10.4)'
      write(*,*)'suggest initial guess D/4......... '
      read(*,*)webrqd
      r=webrqd/tb
      pfac=xx/r
      write(*,21)r
21    format(1x,'The required burn rate =',f6.4)
      write(*,22)pfac
22    format(1x,'The required perimeter factor=',f6.3)
      write(*,25)xx
25    format(1x,'pfac=',f5.4,'/r')
      write(*,24)tb
24    format(1x,'tb  =',f7.2)
      write(*,*)'Are these acceptable? y-yes n-no....... '
      read(*,20)yes
      if(yes.ge.'y'.or.yes.eq.'Y') then
        pwet=pfac*pi*dpe
        wf=webrqd/(d/2)
        j=ath/aport
        ld=l/d
        go to 23
      else
        go to 18
      endif
20    format(a1)
30    format(f12.6)
23    write(*,40)cfc
40    format(1x,'THRUST COEFFICIENT.....................',F12.3)
      write(*,50)etacf
50    format(1x,'ETACF.................................',F12.3)
      write(*,60)isp
60    format(1x,'SPECIFIC IMPULSE,  LBF SEC/LBM.........',f12.3)
      write(*,70)m
70    format(1x,'MASS FLOW RATE,    LBM/SEC.............',f12.3)
      write(*,80)vp
80    format(1x,'PROPELLANT VOLUME,  CU IN..............',f12.1)
      write(*,100)ath
```

APPENDIX

```
100   format(1x,'NOZZLE THROAT AREA, SQ IN..............',f12.3)
      write(*,105)ae
105   format(1x,'NOZZLE EXIT AREA,   SQ IN..............',f12.3)
      write(*,110)dpe
110   format(1x,'EQUIVALENT PORT DIAMETER, IN...........',f12.3)
      write(*,120)aport
120   format(1x,'PORT AREA, SQ IN ......................',f12.3)
      write(*,130)pfac
130   format(1x,'PERIMETER FACTOR.......................',f12.3)
      write(*,140)webrqd
140   format(1x,'REQUIRED WEB, IN.......................',f12.3)
      write(*,150)wf
150   format(1x,'REQUIRED WEB FRACTION..................',f12.3)
      write(*,90)vl
90    format(1x,'VOLUMETRIC LOADING.....................',f12.3)
      write(*,190)ld
190   format(1x,'LENGTH TO DIAMETER RATIO...............',f12.3)
      write(*,170)tres
170   format(1x,'RESIDENCE TIME, MSEC...................',f12.3)
      write(*,180)j
180   format(1x,'THROAT TO PORT AREA RATIO (J FACTOR)...',f12.3)
      write(*,200)r
200   format(1x,'REQUIRED BURNING RATE,   IN/SEC........',f12.3)
      return
      end
```

APPENDIX

```
      subroutine loscf(etacf,tb,ath,pc,alpha,thetaex,erosr,
     *mf,nozle)
c*************************************************************
c                                                            *
c THIS SUBROUTINE CALCULATE THE NOZZLE LOSS COEFFICIENT      *
c (THRUST COEFFICIENT) USING THE METHODS PROPOSED BY AGARD   *
c PROPULSION AND ENERGETICS PANEL FOR METALLIZED PRO-        *
c PELLANTS (Reference: Performance of Rocket with  Metal-    *
c lized propellants, AGARD-AR-230)                           *
c                                                            *
c*************************************************************
c Defination of varibles in this subroutine
c          etacf:    nozzle loss coefficient
c          div  :    percent divergence loss
c          kin  :    percent kinetics loss
c          bl   :    percent boundary layer loss
c          tp   :    percent two-phase loss
c          sub  :    persent submergence loss
c          eros :    percent nozzle erosion loss
c          astar:    nozzle entrance/ nozzle throat area
c          s    :    submergence length/ length of internal motor
c          mf   :    mole fraction of condensed phase in mole/100grm
c          c1,c2,c3,c4,c6 :
c                    constant in emperical relation
c          dp   :    diameter of particle
c          pc   :    chamber pressure
c          ispf :    Isp of frozen equilibrium
c          isps :    Isp of shifting equilibrium
c          nozle:    steel nozzle = 1, ordinary nozzle = 0
c          tb   :    burning time of motor
c          dt   :    throat diameter
c          expn :    nozzle expansion ratio
c          ispi :    Isp with ideal (at design point) nozzle expansion ratio
c          ispm :    Isp with mean nozzle expansion ratio
c
c
      integer expnu,expnl
      real etacf,div,kin,bl,tp,sub,eros,astar,s,mf,dp,pc,isps,ispm,ispf,
     +dt,ispu,ispl,erosr
      common /scratc/ plot(5,100)
      common /socf/ spi(2),cf(2),oex(2)
c calculate the % of nozzle divergence loss
      div = 50 * ( 1 - cos ((alpha + thetaex) / 2 ))
c ispf is the frozen and isps is the shifting specific impulse from
c micropep run at nominal pc
c Transfer specific impulse data from MICROPEP.
      ispf=spi(1)
      isps=spi(2)
c Calculate the kinetic loss
      kin = 33.3 * (1 - ispf/isps)
c calculate % boundary layer loss
c Steel nozzle (1), Ordinary nozzle (0). Enter 0 or
      if (nozle.eq.1) then
         c1 = 0.00506
         c2 = 0.0
      else
         c1 = 0.00365
         c2 = 0.000937
      endif
c
```

APPENDIX

```
c    expansion ratio of the nozzle for shifting equilibrium from
c    micropep with exit pressure=ambient pressure at design altitude
c     read (*,70)expn
      expn=oex(2)
      pi=3.14159
      dt=sqrt(ath/pi)*2
      bl =(cl*pc**0.8/(dt*0.2))*(1+2*exp(-c2*pc**0.8*tb/(dt*0.2)))*(1+0.
     $016*(expn-9.0))
c calculate the two-phase flow loss
c the mole fraction of condensed phase is come from "MICROPEP"
c mean particle diameter calculation by using empirical relation
      dp = 3.39*dt**0.4692
      if (mf.lt.0.09) then
             mf = 0.09
      endif
      if(dt.lt.1.0) then
           c3=9.0
           c5=1.0
           c6=1.0
      else
          if(dt.le.2) then
             c3=9.0
             c5=1.0
             c6=0.8
          else
             if(dp.lt.4) then
                 c3=13.4
                 c5=0.8
                 c6=0.8
             else
                if(dp.le.8) then
                    c3=10.2
                    c5=0.8
                    c6=0.4
                else
                     c3=7.58
                     c5=0.8
                     c6=0.33
                endif
             endif
          endif
      endif
      c4=0.5
      tp=c3*mf**c4*dp**c5/(pc**0.15*expn**0.08*dt**c6)
c calculate the submergence loss(NOT USED IN PRELIMINARY WORK !!)
c the nozzle is assumed to be of external design with no submergence
c loss. s=length of submergence / length ofinternal motor.
c     astar=(pi*d**2/4)/ath
c     sub=0.0684*(pc*expn/astar)**0.8*s**0.4/dt**0.2
c     default to 0.0
      sub=0.0
c calculate the nozzle erosion loss
c calculate the ispm (specific impulse at mean expansion ratio)
      df=dt+erosr*tb*2
      dbar=(dt+df)/2
      expnm=(dt**2/dbar**2)*expn
      expnl=int(expnm)
c Transfer expansion ratio of nozzle and specific impulse from MICROPEP.
      ispl=plot(4,expnl)
      expnu=int(expnm)+1
```

```
      ispu=plot(4,expnu)
      ispm=(expnm-expnl)/(expnu-expnl)*(ispu-ispl)+ispl
c erosion loss calculation
      eros=(1-ispm/isps)*100
      write(*,100)div
  100 format(1x,'PERCENT DIVERGENCE LOSS................... ',f6.3)
      write(*,110)tp
  110 format(1x,'PERCENT TWO PHASE FLOW LOSS............... ',f6.3)
      write(*,120)bl
  120 format(1x,'PERCENT BOUNDARY LAYER LOSS............... ',f6.3)
      write(*,130)kin
  130 format(1x,'PERCENT KINETIC LOSS...................... ',f6.3)
      write(*,140)sub
  140 format(1x,'PERCENT SUBMERGENCE LOSS.................. ',f6.3)
      write(*,150)eros
  150 format(1x,'PERCENT EROSION LOSS...................... ',f6.3)
      etacf=1-(div+tp+bl+kin+sub+eros)/100.0
      write(*,160)etacf
  160 format(1x,'THRUST COEFFICIENT EFFICIENCY............. ',f6.3)
      return
      end
```

APPENDIX

```
      subroutine input2(pfac,aport,web,rr5,lgrain,tb,xx)
c*********************************************************
c.                                                       *
c     THE PRIMARY PURPOSE OF THIS SUBROUTINE IS TO ES-    *
c     TABLISH THE DATA FILE FOR THE CSD "GRAINS" PROGRAM  *
c                                                         *
c*********************************************************
      character*80 tape5
      character*1 yes
      real lgrain,lspoke,kspoke,RR5
      integer nmax,nspoke,ibb,iprnt
      write(*,*)'The following will be the data input for '
      write(*,*)'spoked star grain designs with radial sides '
      open(4,file='grainin.dat',status='unknown')
      write(*,*)'Run DESIGN-1 mode or BURNBACK-0? Type 1 or 0... '
      read(*,*)i
      if(i.eq.1)then
11      write(*,40)web
14      r=web/tb
        pfac=xx/r
        write(*,13)r
13      format(1x,'Required burning rate, IN/SEC............... ',f12.3)
        write(*,50)pfac
        write(*,*)'Do you want to change the web?...y/n........... '
        read(*,5)yes
        if(yes.eq.'Y'.or.yes.eq.'y')then
          write(*,*)'Input the required web, IN................. '
          read(*,*)web
          go to 14
        endif
        write(*,*)'Max number of spokes?.(I3).................... '
        read(*,*)nmax
        write(*,*)'Initial number of spokes?.(I3)................ '
        read(*,*)nspoke
        ibb=1
        iprnt=0
        write(4,10)nmax,nspoke,ibb,iprnt
        write(*,30)aport
   30   format(1x,'Required port area, SQ IN..................... ',f12
     *.3)
        write(*,40)web
   40   format(1x,'Initial web thickness in IN................... ',f12
     *.3)
        write(*,50)pfac
   50   format(1x,'Required perimeter factor .................... ',f12
     *.3)
        write(*,*)'Input Kspoke - Abstract multiplier on '
        write(*,*)'PHI2 - if KSPOKE=0.0 PHI2=0.0 where:'
        write(*,*)'PHI2=ATAN(KSPOKE*WEB/RR3/COS(PHI3))   (0.0-1.0).... '
        read(*,*)kspoke
        write(*,60)rr5
   60   format(1x,'Grain outer radius,INCHES......................',f12
     *.3)
        write(4,20)aport,web,pfac,kspoke,rr5
        write(*,*)'Input initial grain inner radius, INCHES (F7.3) '
        read(*,*)ri1
        write(*,*)'Input initial value of SR1 - spoke'
        write(*,*)'top corner radius in IN (Default=0.25)......... '
        read(*,*)sri1
        write(*,*)'Input initial value of SR2 - spoke'
```

```
      write(*,*)'base coner radius in IN (Default=0.25)......... '
      read(*,*)sri2
      write(4,20)ri1,sri1,sri2
      write(*,*)'Input increment on SR1 in IN - cannot be 0.0... '
      read(*,*)del1
      write(*,*)'Input increment on RR1 in IN - negative value.. '
      read(*,*)del2
      write(*,*)'Input increment on NSPOKE Default = 1.0........ '
      read(*,*)del3
      write(*,*)'Input increment on SR2 in IN - cannot be 0.0... '
      read(*,*)del4
      write(4,20)del1,del2,del3,del4
      write(*,*)'Input tolerance on grain angle sum total - %'
      write(*,*)'of 360.(0.0-1.0)................................ '
      read(*,*)cirtol
      write(*,*)'Input tolerance on port area - % of Aport '
      write(*,*)'(0.0-1.0).................................. '
      read'* *)pttol
      write(*,*)'Input tolerance on perimeter - % of perimeter.. '
      write(*,*)'(0.0-1.0)...................................... '
      read(*,*)pertol
      write(4,20)cirtol,pttol,pertol
      write(*,*)'Input Max of SR1 in IN .(F10.5)................ '
      read(*,*)fin1
      write(*,*)'Input Min of RR1 in IN .(F10.5)................ '
      read(*,*)fin2
      write(*,*)'Input Max of SR2 in IN .(F10.5)................ '
      read(*,*)fin4
      write(*,*)'Input Max of web in IN .(F10.5)................ '
      read(*,*)finweb
      write(*,*)'Input increment for web in IN.(F10.5).......... '
      read(*,*)webstp
      write(4,20)fin1,fin2,fin4,finweb,webstp
    else
      nmax=0
      write(*,*)'Number of spokes?...(I3)...................... '
      read(*,*)nspoke
      ibb=-1
      write(*,*)'Do you want optional output? y-yes, n-no....... '
      read(*,5)yes
      if(yes.eq.'Y'.or.yes.eq.'y')then
        iprnt=-1
      else
        iprnt=0
      endif
      write(4,10)nmax,nspoke,ibb,iprnt
      write(*,70)
70    format(1x,'Input initial port area, SQ IN..(F10.5)......... ')
      read(*,*)aport
      write(*,80)
80    format(1x,'Input initial web thickness, IN.(F10.5)......... ')
      read(*,*)web
      write(*,90)
90    format(1x,'Input the perimeter factor..(F10.5).............. ')
      read(*,*)pfac
      kspoke=0.0
      write(*,60)rr5
      write(4,20)aport,web,pfac,kspoke,rr5
      write(*,*)'Input initial RR1, INCHES.(F10.5)................ '
      read(*,*)ri1
```

```
          write(*,*)'Input initial value of SR1 (Default=0.25 IN.).... '
          read(*,*)sri1
          write(*,*)'Input initial value of SR2 (Default=0.25 IN.)..... '
          read(*,*)sri2
          write(4,20)ri1,sri1,sri2
          del1=0.0
          del2=0.0
          del3=0.0
          del4=0.0
          write(4,20)del1,del2,del3,del4
          cirtol=0.0
          pttol=0.0
          pertol=0.0
          write(4,20)cirtol,pttol,pertol
          fin1=0.0
          fin2=0.0
          fin4=0.0
          finweb=0.0
          webstp=0.0
          write(4,20)fin1,fin2,fin4,finweb,webstp
          write(*,*)'Input grain length, INCHES (F10.5)............ '
          read(*,*)lgrain
          write(*,*)'Input initial PHI2, DEGREES (F10.5)........... '
          read(*,*)dphi2
          write(*,*)'Input initial PHI4, DEGREES (F10.5)............ '
          read (*,*)dphi4
          write(4,20)lgrain,dphi2,dphi4
          write(*,*)'Input # of steps in decimal format'
          write(*,*)'Input # of burnback steps for SR1>0............. '
          read(*,*)step1
          write(*,*)'Input # of burnback steps for PHI2>0 & SR1=0.... '
          read(*,*)step2
          write(*,*)'Input # of burnback steps for LSPOKE=0 & PHI2>0. '
          read(*,*)step3
          write(*,*)'Input # of burnback steps for '
          write(*,*)'triangular spoke burnback...................... '
          read(*,*)step4
          write(*,*)'Input # of burnback steps for '
          write(*,*)'remaining fuel burnout......................... '
          read(*,*)step5
          write(4,20)step1,step2,step3,step4,step5
        endif
 5      format(a1)
10      format(4i3)
20      format(5f10.5)
        close(4,status='keep')
        return
        end
```

REFERENCES

1. Haymes, W.G., Williamson, J.E., McClendon, S. E., and Brooks, W. T., Solid Rocket Motor Design Automation Technology, AGARD Conference Proceeding No. 259, April 1979.

2. National Aeronautics and Space Administration NASA SP-8064, Solid Propellant Selection and Characterization, June 1971.

3. Class Notes: AE-4452, "Missile Propulsion" by Netzer, D. W., unpublished, March 1987.

4. Cruise, D. R., Naval Weapons Center, NWC-TP-6037, Theoretical Computations of Equilibrium Computations, Thermodynamic Properties, and Performance Characteristics of Propellant Systems, April 1979.

5. National Aeronautics and Space Administration NASA SP-8076, Solid Propellant Grain Design and Internal Ballistics, March 1972.

6. Advisory Group for Aerospace Research & Development AGARD-AR-230, Propulsion and Energetics Panel Working Group 17 on Performance of Rocket Motors with Metallized Propellants, September 1986.

7. National Aeronautics and Space Administration NASA SP-8039, Solid Rocket Motor Performance Analysis and Prediction, May 1971.

8. Coenen, Y. G., Naval Weapons Center NWC-TM-4757, Analyst's Manual and User's Guide for "FLYIT3", A Three-Degree-of-Freedom (3-DOF) Trajectory Simulation Program, April 1982.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5100	2
3.	Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 93943-5100	1
4.	Professor D. W. Netzer, Code 67Nt Department of Aeronautics Naval Postgraduate School Monterey, California 93943-5100	2
5.	Library Chinese Naval Academy Tso-yin, Taiwan Republic of China	3
6.	LCDR Yuan, Chung-I Fl. 4, No. 19, 12th Alley, 139th Lane, Sec. 2, Pei-Hsin Rd, Taipei, Taiwan Republic of China	17
7.	Marie B. Hashimoto Department of Aeronautics, Code 67 Naval Postgraduate School Monterey, California 93943-5100	1
8.	Deputy Chief of Naval Operations (Air Warfare) OP-05 Pentagon Washington, D.C. 20361	1
9.	F. Zarlingo Code 32403 Naval Weapons Center China Lake, California 93555	1

END
DATE
FILMED
6-1988
DTIC